JetOS95

User Manual

www.korenix.com

Copyright Notice

Copyright© 2011 Korenix Technology Co., Ltd.
All rights reserved.
Reproduction without permission is prohibited.

Information provided in this manual is intended to be accurate and reliable. However, the original manufacturer assumes no responsibility for its use, or for any infringements upon the rights of third parties that may result from its use.
The material in this document is for product information only and is subject to change without notice. While reasonable efforts have been made in the preparation of this document to assure its accuracy, Korenix assumes no liabilities resulting from errors or omissions in this document, or from the use of the information contained herein.
Korenix reserves the right to make changes in the product design without notice to its users.

Acknowledgments

Korenix is a registered trademark of Korenix Technology Co., Ltd.
All other trademarks or registered marks in the manual belong to their respective manufacturers.

Table of Content

Copyright Notice 2
Acknowledgments 2
Table of Content 3
Chapter 1 Overview 6
1-1 Applied Models 6
Chapter 2 Getting Start 7
2-1 System login 7
2-1-1 Serial Console 7
2-1-2 Telnet Console 8
2-1-3 SSH Console 9
2-2 Configure Ethernet Interface 10
2-2-1 Modifying Network Settings with the Serial Console 10
2-2-2 Static and Dynamic IP address 12
2-2-3 Modifying Network Settings over the Network 12
2-3 Test Program Developing – Hello.c 13
2-3-1 Installing the Toolchain (Linux) 14
2-3-2 Compiling Hello.c 14
2-3-3 Uploading “test” to JetBox9500 and Running the Program 15
Chapter 3 System Feature 17
3-1 System Version 17
3-2 Enable/Disable Daemons 17
3-3 Setting Run-Level 19
3-4 Setting System Time 19
3-5 Adjust System Time 20
3-5-1 NTP Client 20
3-5-2 NTP Server 21
3-6 Cron Daemon (Schedule jobs) 21
3-7 Connect Peripherals 22
Chapter 4 Network Feature 24
4-1 Telnet 24
4-2 sshd 24
4-3 FTP 25
4-4 DNS 25
4-5 Web Server 26
4-6 IPTABLES 27
4-7 NAT 29
4-8 Dail-up Service 30

4-9 PPPoE 30
4-10 NFS 32
4-11 E-Mail 32
4-12 SNMP 33
4-13 OpenVPN 33
4-14 OpenSwan 34
4-15 PPTP Client 35
4-16 PPTP Server 37
4-17 Routing -- Zebra Daemon 37
4-18 Ser2net 40
4-19 fastpath 41
4-20 keepalived 42
4-21 PTPD 47
4-22 PTPDv2 48
Chapter 5 Korenix Feature 51
5-1 Ethertool 51
5-2 Vconfig 52
5-3 Diocfg 53
5-4 Poecfg 54
Chapter 6 Programmer's Guide 55
6-1 Toolchain Introduction 55
6-1-1 Compiling Applications and Libraries 56
6-1-2 Tools Available in the Host Environment 56
6-2 Device API 57
6-3 RTC 57
6-4 Watch Dog Timer 58
Chapter 7 Appendix 60
7-1 Firmware Upgrade 60
7-2 JetView 60
7-2-1 Overview 60
7-2-2 JetView for JetBox 9500 60
7-3 USB Driver for GSM modems 61
7-4 Change RS485 mode on JetCard 1608 61
7-5 Software Specification 62
7-6 Busybox command 64
7-7 Ser2net Manual 69
7-8 Revision history 73
7-9 Customer Service 73

Chapter 1 Overview

The advantage of adopting Korenix JetBox series is ready-to-use. Korenix is devoted to improve the usability of embedded computer in industrial domain. Besides operating system, Korenix provides device drivers, protocol stacks, system utilities, supporting services and daemons to make system integration simple. Further, Korenix provides application development toolkits for users to build up their own applications easily.

The stylish JetBox 9500 series is an industrial layer-3 router with power-over-Ethernet technology and Linux computing. It is a gateway to connect different network groups (Ethernet, fieldbus, serial or IO control) in a complex networking architecture and manage peripherals at the front-end site through its Linux programs or Java applications. It is reliable (network redundancy, system recovery) and robust (passive cooling, protected against the dusts and spills, shock & vibration resistance) to adopt in severe industrial vertical markets, such as transportation, substation, or hazardous environment.

1-1 Applied Models

JetBox9500 series
JetBox9400 series
JetBox5400 series
JetCard5400

Note: SW features might be different according to different products

Chapter 2 Getting Start

2-1 System login

Users can enter the JetBox Linux environment via the user name: root and no password is required.

login : root
password : (none)

2-1-1 Serial Console

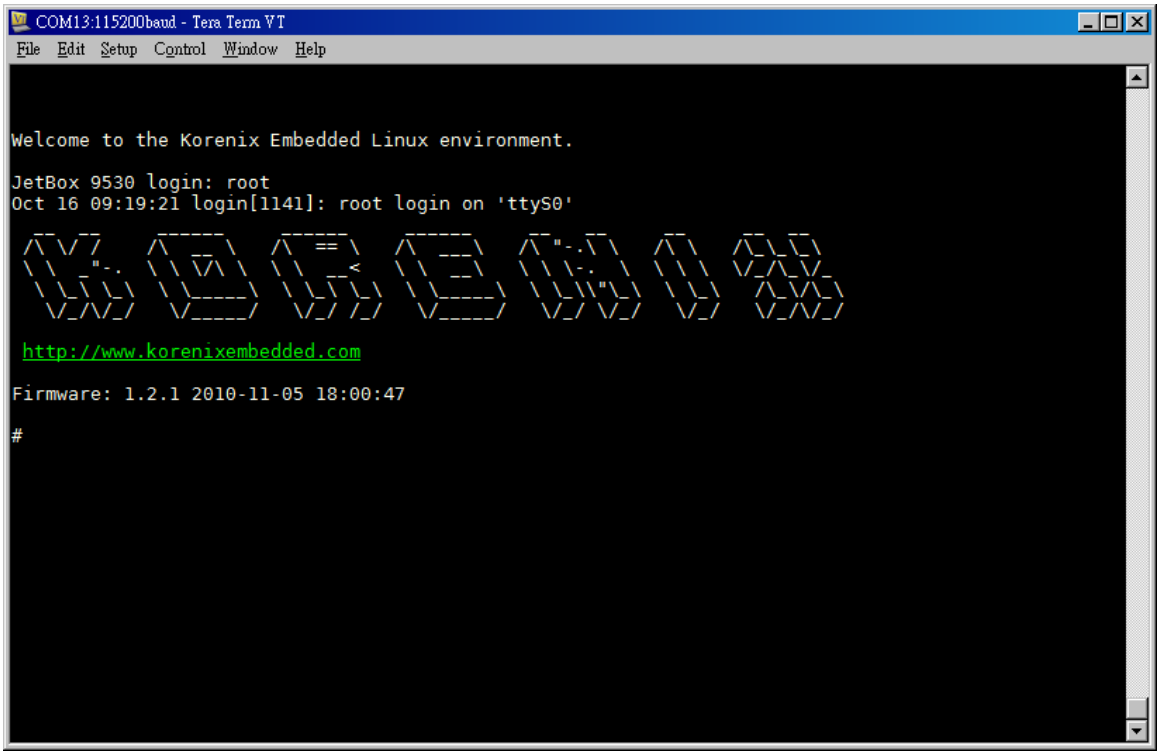
The serial console port gives users a convenient way of connecting to JetBox console utility. This method is particularly useful when using JetBox for the first time. The signal is transmitted over a direct serial connection, so you do not need to know either of JetBox's IP address in order to connect to the serial console utility.

Use the serial console port settings shown below.

Baud rate	115200bps
Parity	None
Data bits	8
Stop bits	1
Flow Control	None

Serial console port setting

Once the connection is established, the following windows will open.


COM13:115200baud - Tera Term VT
File Edit Setup Control Window Help
Welcome to the Korenix Embedded Linux environment.
JetBox 9530 login: root
Oct 16 09:19:21 login[1141]: root login on 'ttyS0'
http://www.korenixembedded.com
Firmware: 1.2.1 2010-11-05 18:00:47
#

Serial console screen

To log in, type the Login name and password as requested. The default values as following.

Login: root

Password: none

2-1-2 Telnet Console

If you know IP addresses and netmasks, then you can use Telnet to connect to JetBox9500. The default IP address and Netmask for each port is given below:

	Default IP address	Netmask
LAN	192.168.10.1	255.255.255.0
WAN	DHCP	

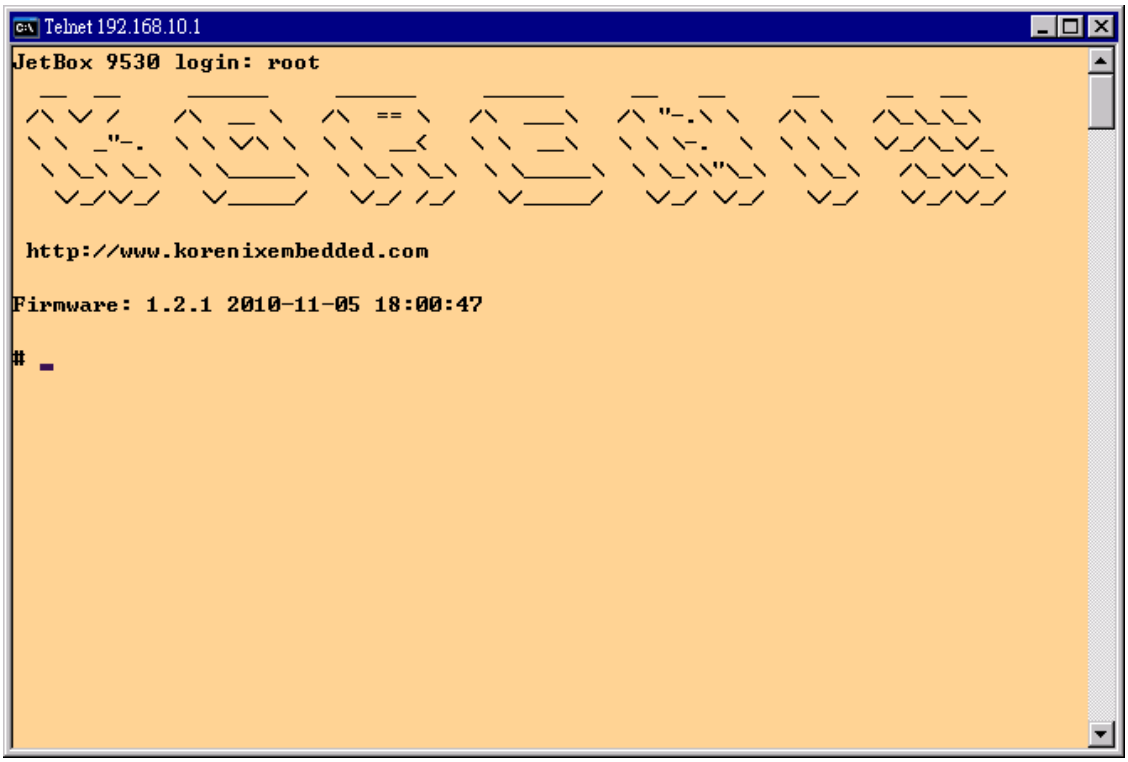
Default IP address and Netmask

Use a cross-over Ethernet Cable to connect directly from your PC to JetBox 8100. You should first modify your PC's IP address and netmask so that your PC is on the same subnet as JetBox.

To connect to a hub or switch connected to your local LAN, use a straight-through Ethernet cable. The default IP address and netmasks are shown above. To login, type the Login name and password as requested. The default values as following:

Login: root

Password: none



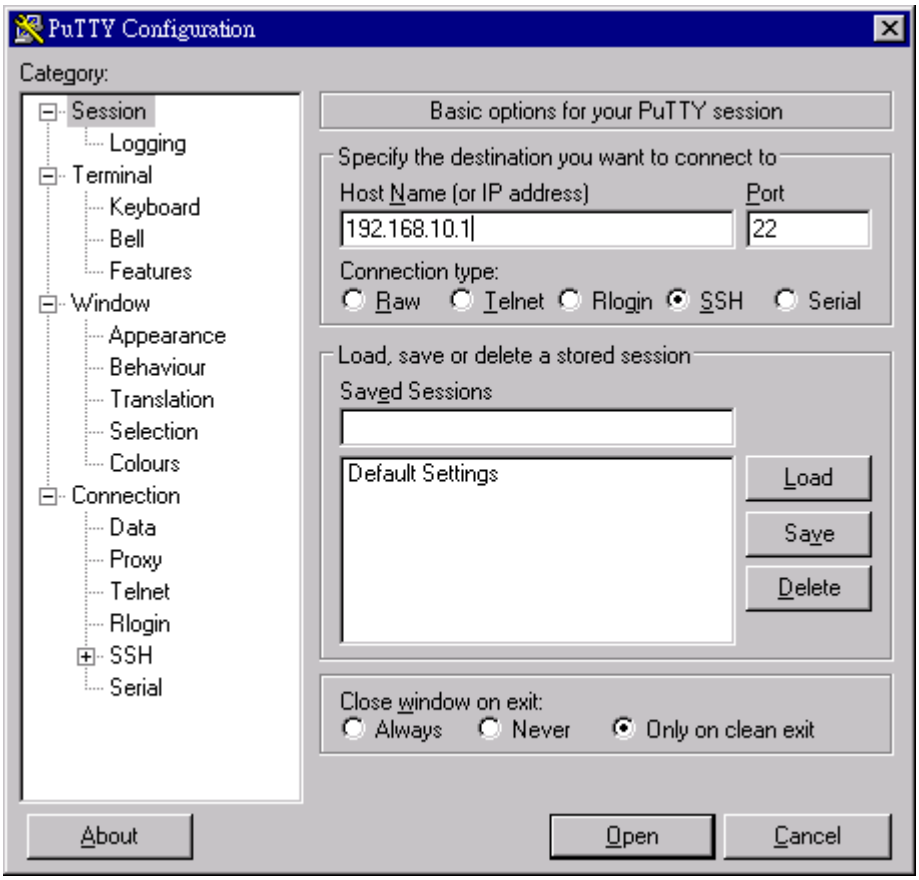
Telnet console screen

You can proceed with the configuration of JetBox9500's network settings when you reach the bash command shell. Configuration instructions are given in the next section.

2-1-3 SSH Console

Start from firmware v1.3, JetBox9500 supports an SSH Console to offer users with better security options.

Click on the link putty to download PuTTy(freeware) and set up an SSH console for JetBox 9500 in a Windows environment. The following figure shows an example of the configuration that is required.



Windows PuTTy setting

2-2 Configure Ethernet Interface

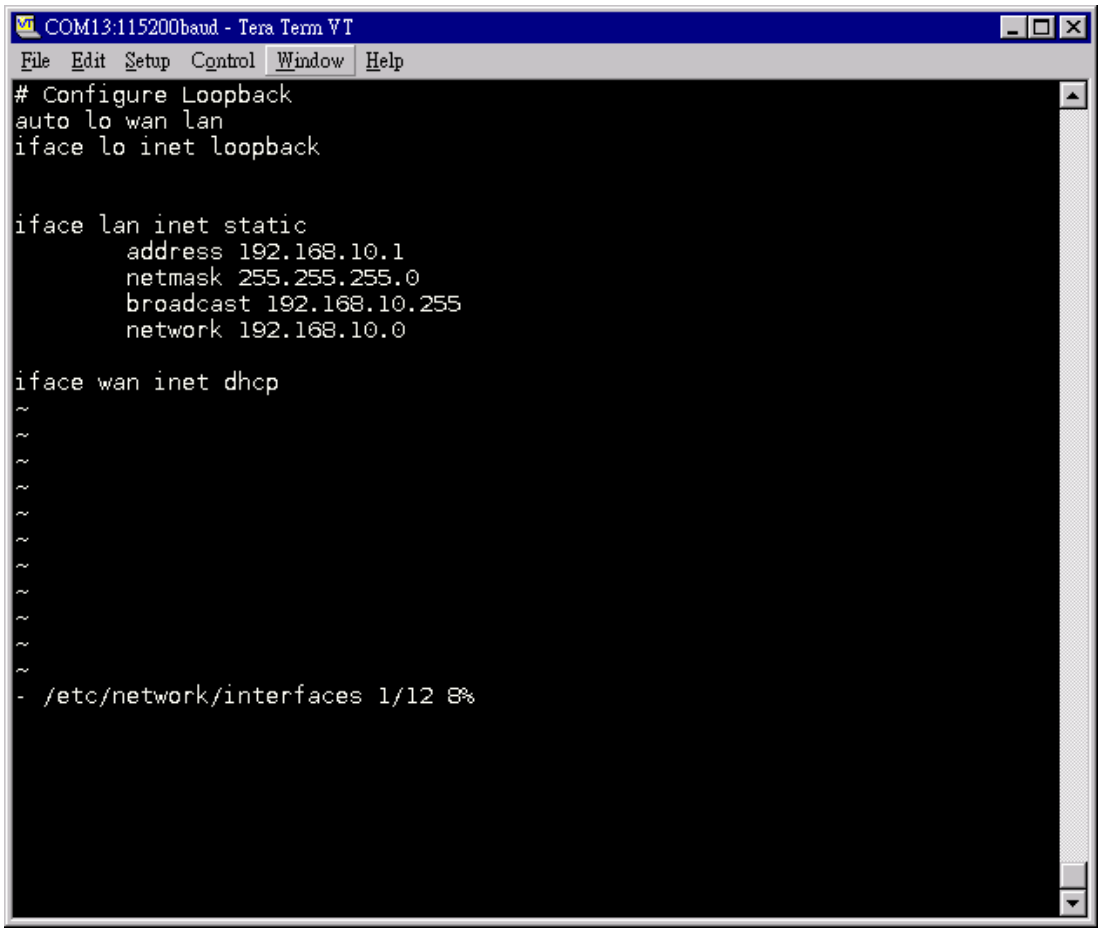
JetBox9500 network setting can be modified with the serial console, or over the network.

2-2-1 Modifying Network Settings with the Serial Console

In this section, we use the serial console to modify JetBox9500's network settings

- **Change Network Configuration**

  Follow the instructions given in a previous section to access JetBox9500's Console Utility via the serial Console port, and then type 'vi /etc/network/interfaces' to edit network configuration file with vi editor.



Edit Network configuration file

- **Static and Dynamic IP address**

Static IP address:

As shown in below, 4 fields must be modified: **address**, **netmask**, **broadcast** and **network**. The default IP addresses are 192.168.10.1.

```
iface lan inet static
        address 192.168.10.1
        netmask 255.255.255.0
        broadcast 192.168.10.255
        network 192.168.10.0
```

Dynamic IP addresses:

By default, the Jetbox9500 is configured for “static” IP addresses on LAN port and

dhcp on WAN interface. To configure LAN port to request an IP address dynamically, remove the original settings and add the following line.

```
iface lan inet dhcp
```

Default setting for LAN port	Default setting for WAN port
iface lan inet static address 192.168.10.1 netmask 255.255.255.0 broadcast 192.168.10.255 network 192.168.10.0	iface wan inet dhcp

Default Gateway:

When static IP Address setting is used, add a default gateway is to set another keyword “gateway”. For example:

```
iface wan inet static
        address 192.168.1.2
        netmask 255.255.255.0
        broadcast 192.168.1.255
        network 192.168.1.0
        gateway 192.168.1.1    # This will set default gateway to
                               #192.168.1.1 on WAN port
```

2-2-2 Static and Dynamic IP address

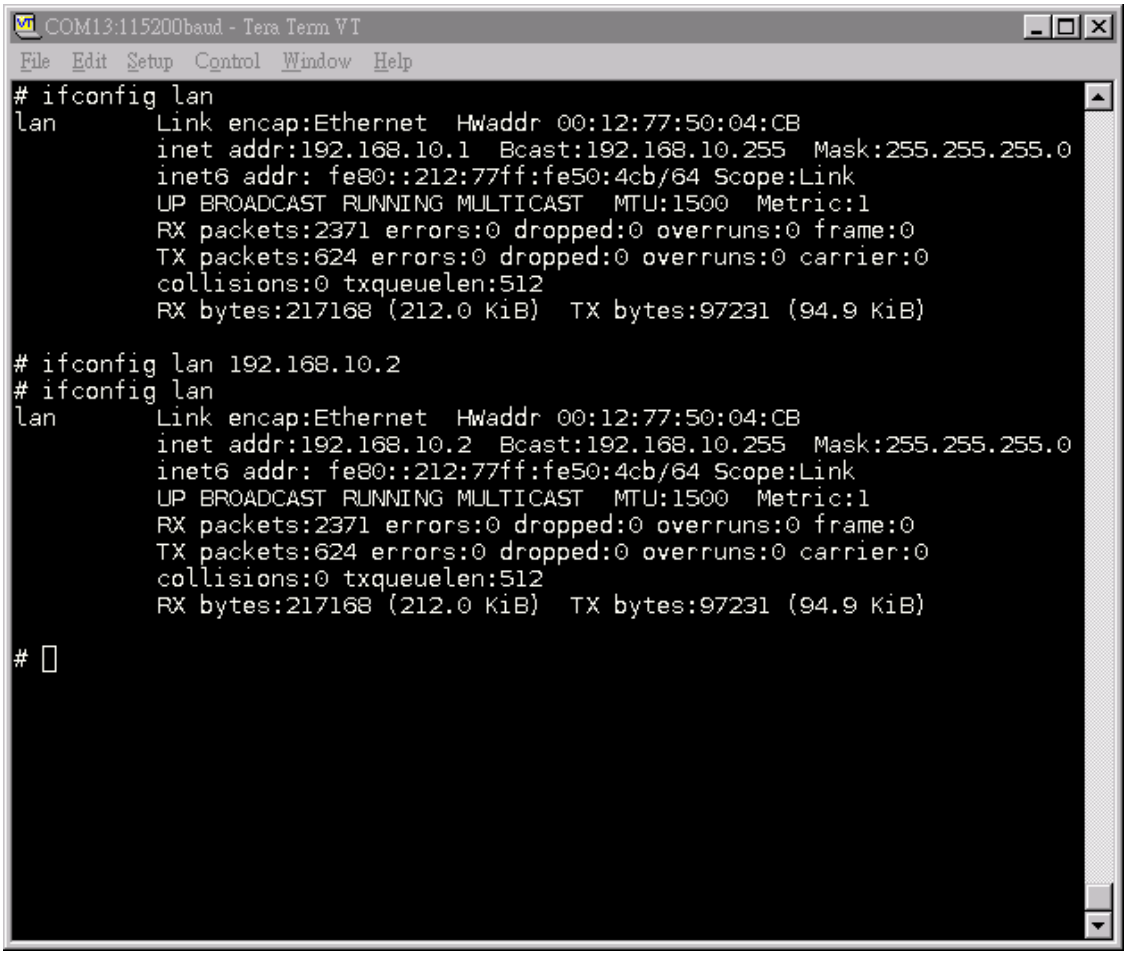
After the /etc/network/interfaces file have been modified, issue the following command to apply the network settings immediately:

```
/etc/init.d/network restart
```

2-2-3 Modifying Network Settings over the Network

Same the previous section, IP settings can be modified over the network, too. There is another way to change the IP address without modifying the file /etc/network/interfaces, but the new settings will not be saved to the flash disk.

For example, type the command **#ifconfig lan 192.168.10.2** to change the IP address of LAN interface to 192.168.10.2.


COM13:115200baud - Tera Term VT
File Edit Setup Control Window Help
ifconfig lan
lan Link encap:Ethernet HWaddr 00:12:77:50:04:CB
inet addr:192.168.10.1 Bcast:192.168.10.255 Mask:255.255.255.0
inet6 addr: fe80::212:77ff:fe50:4cb/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:2371 errors:0 dropped:0 overruns:0 frame:0
TX packets:624 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:512
RX bytes:217168 (212.0 KiB) TX bytes:97231 (94.9 KiB)
ifconfig lan 192.168.10.2
ifconfig lan
lan Link encap:Ethernet HWaddr 00:12:77:50:04:CB
inet addr:192.168.10.2 Bcast:192.168.10.255 Mask:255.255.255.0
inet6 addr: fe80::212:77ff:fe50:4cb/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:2371 errors:0 dropped:0 overruns:0 frame:0
TX packets:624 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:512
RX bytes:217168 (212.0 KiB) TX bytes:97231 (94.9 KiB)
#

Network Setting over the Network

2-3 Test Program Developing – Hello.c

In this section, we use the standard “Hello” programming example to illustrate how to develop a program for the JetBox9500. In general, program development involves the following seven steps.

Step 1:

Connect the JetBox9500 to a Linux PC.

Step 2:

Install Toolchain on the Linux PC.

Step 3:

Set the cross compiler and PATH environment variables.

Step 4:

Code and compile the program.

Step 5:

Download the program to the JetBox9500 via FTP.

Step 6:

Debug the program

- If bugs are found, return to Step 4.
- If no bugs are found, continue with Step 7.

Step 7:

Back up the user directory (distribute the program to additional JetBox9500 units if needed).

2-3-1 Installing the Toolchain (Linux)

The Linux Operating System must be pre-installed in the PC before installing the JetBox9500 Toolchain. Fedora core or compatible versions are recommended. The Toolchain requires approximately 208 MB of hard disk space on your PC. The JetBox9500 Toolchain is included with JetBox9500 SDK, which can download from Korenix web site. To install the Toolchain, it is simply a matter of extracting a tarball at the proper place:

```
tar jxvf jetbox9500-toolchains.tar.bz2 -C /
```

NOTE

- To install the Toolchain, you must grant root permission.
- Toolchains used to not be relocatable! You must install them in the location they were built for.

Install toolchain is simple, just extract to the /opt directory with the command.
We have include the /opt dir in the tarball, so just uncompress it to /. Next, set up the PATH environment variable, add the toolchain path with export command. The PATH is a variable of your current SHELL. This is also the default search path of Linux command.

```
export PATH=/opt/korenix/toolchains/armeb-linux/usr/bin/:$PATH
```

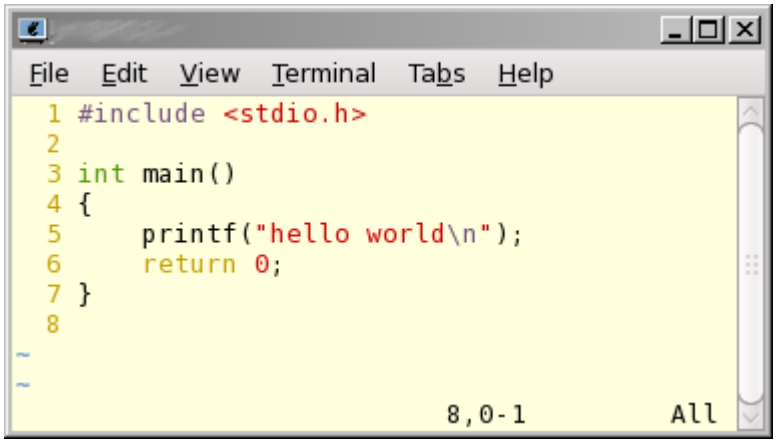
Setting the PATH allows you to run the compiler from any directory.

2-3-2 Compiling Hello.c

If you have been compiling a program on X86, then you will find the only difference is the GCC command is start with **armeb-linux-**.
That is because we want to differentiate with the stand GCC compiler, and the prefix also tell

you – it is for arm, big-endian platform program.

Below is a simple hello.c program:



To compile the helo.c, use our Toolchain to compile the hello.c:

```
armeb-linux-gcc hello.c -o helloworld
```

The output executable file is the *helloworld*.

2-3-3 Uploading "test" to JetBox9500 and Running the Program

Use the following command to upload helloworld to the JetBox9500 via FTP.

1. From the PC, type:

 # ftp xxx.xxx.xxx.xxx

2. Use *bin* command to set the transfer mode to Binary mode, and the put command to initiate the file transfer:

ftp> bin

ftp> put test

```
[root@server ~]# ftp 192.168.10.1
Connected to 192.168.10.1 (192.168.10.1).
220 ProFTPD 1.3.1 Server (ProFTPD TEST Installation) [::ffff:192.168.10.1]
Name (192.168.10.1:root): root
331 Password required for root
Password:
230 User root logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd /home/
250 CWD command successful
ftp> put helloworld
local: helloworld remote: helloworld
```

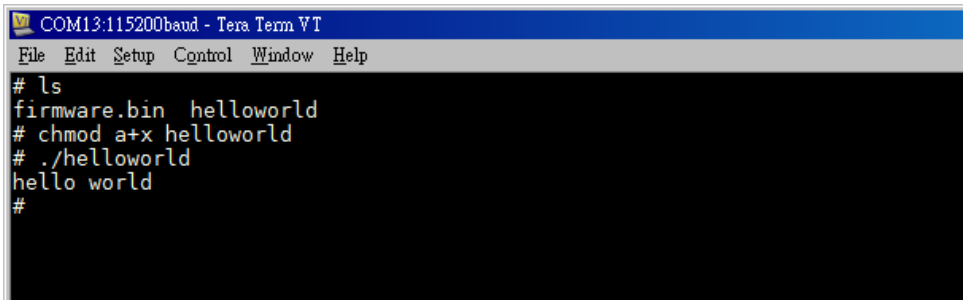
```
227 Entering Passive Mode (192,168,1,176,19,6).
150 Opening BINARY mode data connection for helloworld
226 Transfer complete
4455 bytes sent in 4.1e-05 secs (108658.54 Kbytes/sec)
ftp> exit
221 Goodbye.
```

3. From the JetBox9500 console, type:

 # chmod +x helloworld

 # ./helloworld

The word hello world will be printed on the screen.



Please read Chapter 6 for more detail about Toolchain.

Chapter 3 System Feature

This chapter includes information about version control, deployment, updates, and peripherals. The information in this chapter will be particularly useful when you need to run the same application on several JetBox units.

3-1 System Version

To determine the hardware capability of your JetBox, and what kind of software functions are supported, check the version numbers of your JetBox's hardware, kernel, and user file system. Contact Korenix to determine the hardware version. You will need the Production S/N (Serial number), which is located on the JetBox9500's back label.

To check the **firmware** version, type:

```
# version
```

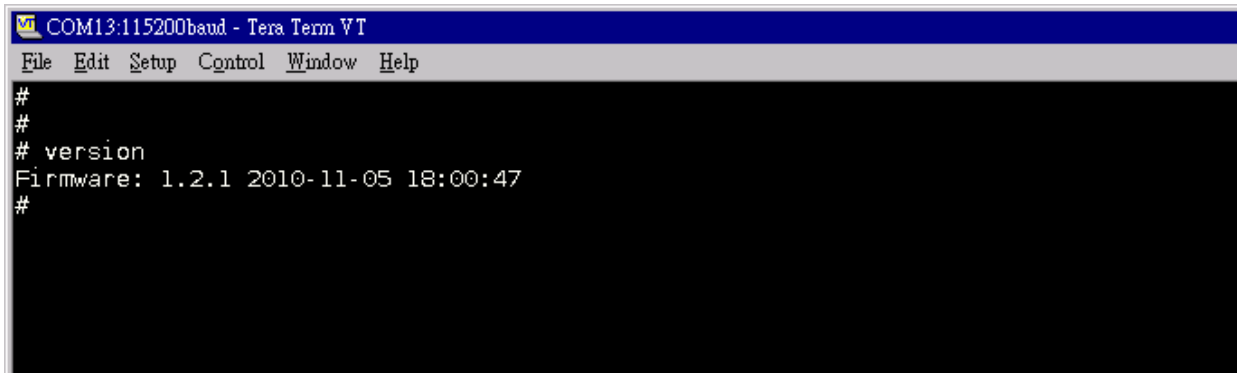


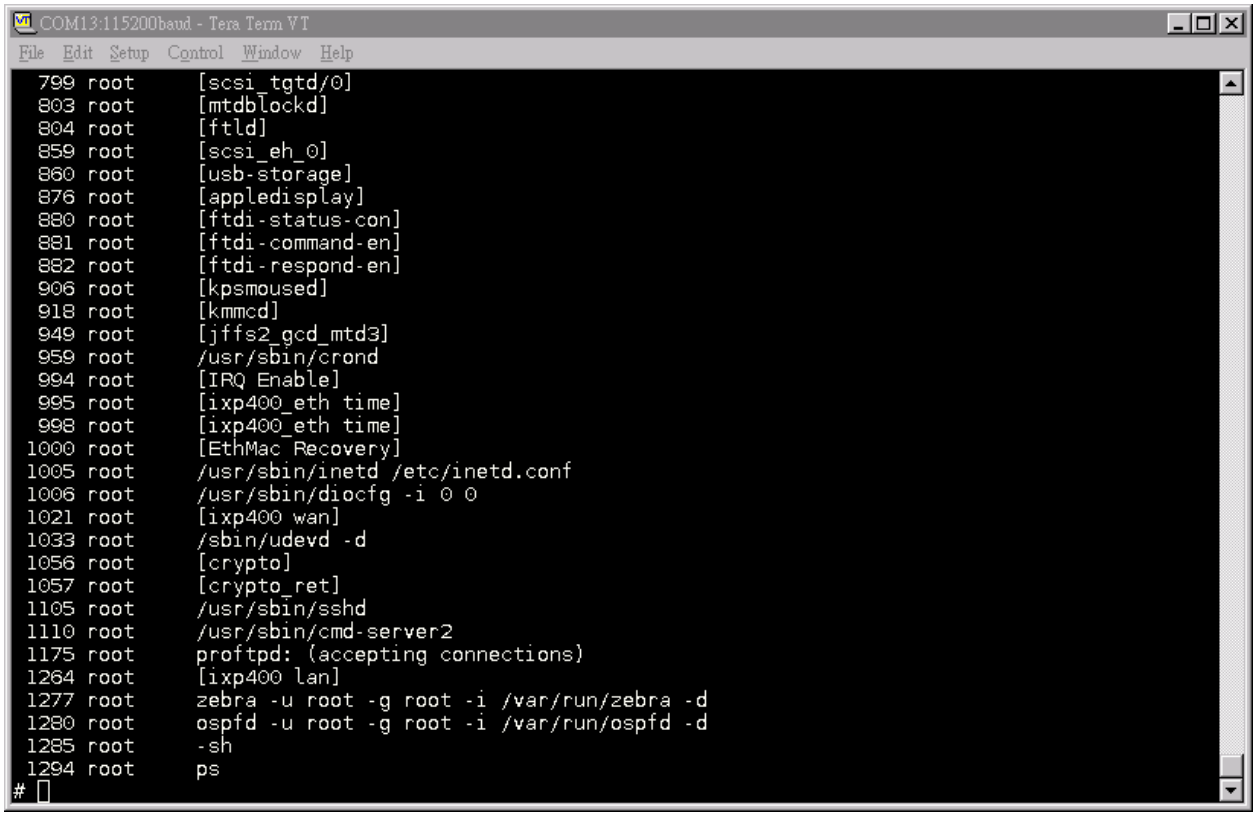
Figure 3-1 Firmware version

3-2 Enable/Disable Daemons

The following daemons are enabled when the JetBox 9500 boots up for the first time.

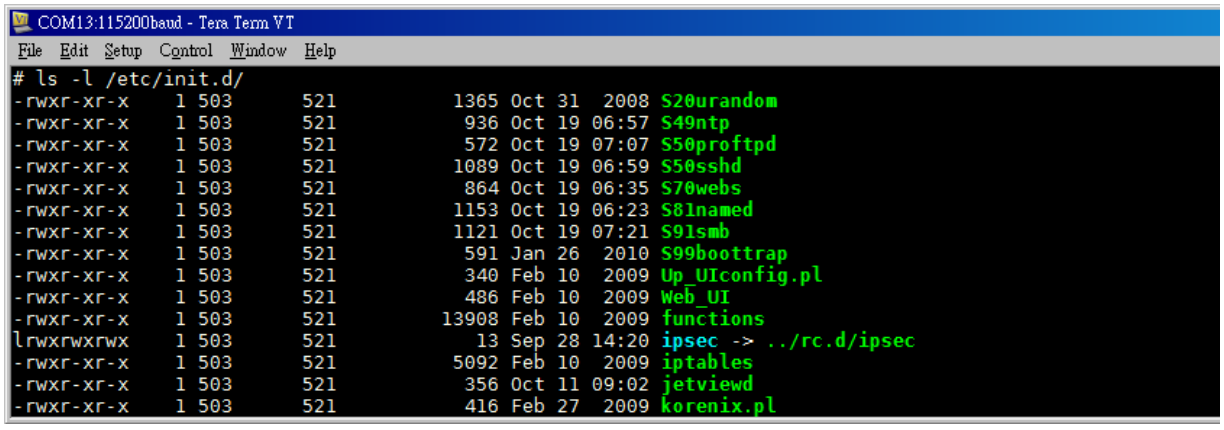
Service name	Description
inetd	internet daemons
telnetd	telnet daemon
sshd	secure shell daemon
crond	schedule jobs daemon

Type the command "**ps**" to list all processes currently running.

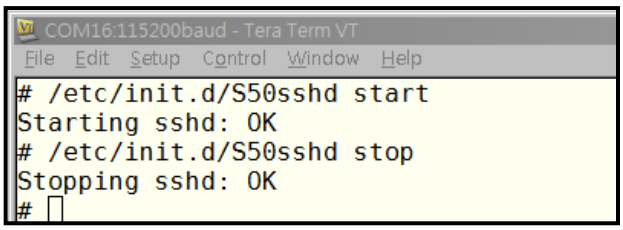


daemons status

The /etc/init.d directory is the repository for all available init scripts. The /etc/rc<n>.d/ only contains links to the /etc/init.d/.


COM13:115200baud - Tera Term VT
File Edit Setup Control Window Help
ls -l /etc/init.d/
-rwxr-xr-x 1 503 521 1365 Oct 31 2008 S20urandom
-rwxr-xr-x 1 503 521 936 Oct 19 06:57 S49ntp
-rwxr-xr-x 1 503 521 572 Oct 19 07:07 S50proftpd
-rwxr-xr-x 1 503 521 1089 Oct 19 06:59 S50sshd
-rwxr-xr-x 1 503 521 864 Oct 19 06:35 S70webs
-rwxr-xr-x 1 503 521 1153 Oct 19 06:23 S81named
-rwxr-xr-x 1 503 521 1121 Oct 19 07:21 S91smb
-rwxr-xr-x 1 503 521 591 Jan 26 2010 S99boottrap
-rwxr-xr-x 1 503 521 340 Feb 10 2009 Up_UIconfig.pl
-rwxr-xr-x 1 503 521 486 Feb 10 2009 Web_UI
-rwxr-xr-x 1 503 521 13908 Feb 10 2009 functions
lrwxrwxrwx 1 503 521 13 Sep 28 14:20 ipsec -> ../rc.d/ipsec
-rwxr-xr-x 1 503 521 5092 Feb 10 2009 iptables
-rwxr-xr-x 1 503 521 356 Oct 11 09:02 jetviewd
-rwxr-xr-x 1 503 521 416 Feb 27 2009 korenix.pl

Here is an example of starting and stopping the ssh daemon:



You can start the service by adding the first argument start, and stop the service by adding the

first argument stop

3-3 Setting Run-Level

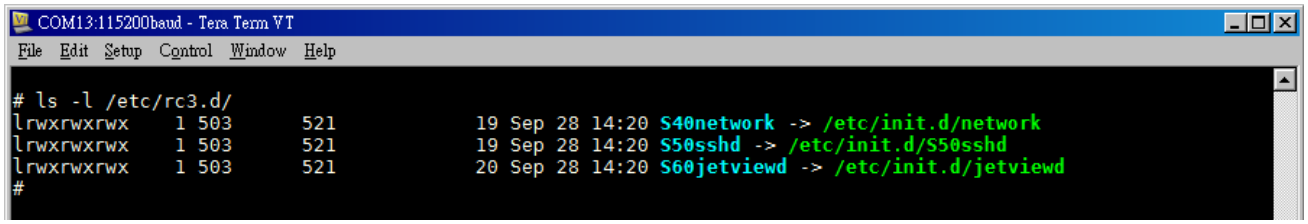
The “run level” setting can be configuring at the file **/etc/inittab**. Conventionally, seven runlevels exist, numbered from zero to six; though up to ten, from zero to nine, may be used. You can change the number to boot JetBox into different runlevel.

```
id:3:initdefault:
```

In this example, runlevel 3 is the default and the scripts that define runlevel 3 can be found in /etc/rc3.d/.

NOTE

Scripts are run in file name lexical order. Just use ls -l to find out the start order.


COM13:115200baud - Tera Term VT
File Edit Setup Control Window Help
ls -l /etc/rc3.d/
lrwxrwxrwx 1 503 521 19 Sep 28 14:20 S40network -> /etc/init.d/network
lrwxrwxrwx 1 503 521 19 Sep 28 14:20 S50sshd -> /etc/init.d/S50sshd
lrwxrwxrwx 1 503 521 20 Sep 28 14:20 S60jetviewd -> /etc/init.d/jetviewd
#

3-4 Setting System Time

The JetBox9500 has two time settings. One is the system time, and the other is the RTC (Real Time Clock) time kept by the JetBox9500’s hardware.

Use the **#date** command to query the current system time or set a new system time.

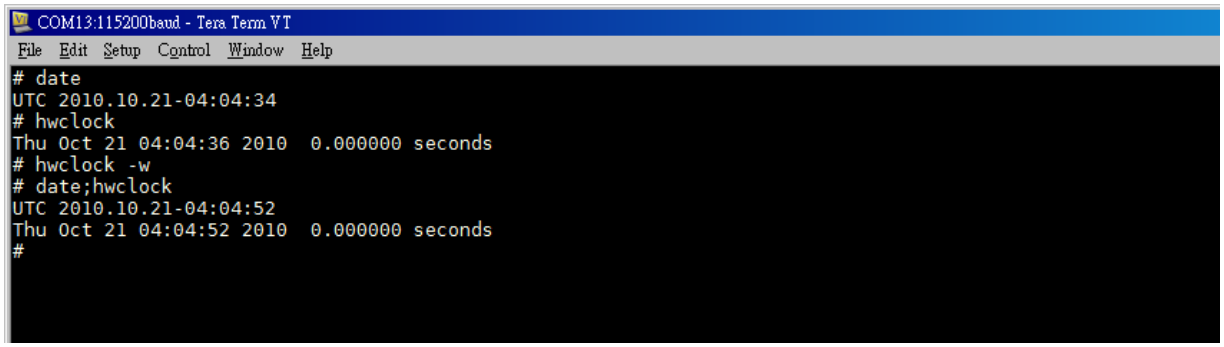
```
#date MMDDhhmmYYYY
MM = Month
DD = Date
hhmm = hour and minute
YYYY = Year
```

Use **#hwclock** to query the current RTC time

Use the following command to set system time from hardware clock:

```
#hwclock -s
```

The following figure illustrates how to update the system time and set the RTC time.



Setting the Time Manually

3-5 Adjust System Time

If you only wish to synchronize your clock when the device boots up, you can use **ntpdate**. This may be appropriate for some devices which are frequently rebooted and only require infrequent synchronization.

Using ntpdate at boot time is also a good idea for devices that run ntpd. The ntpd program changes the clock gradually, whereas ntpdate sets the clock, no matter how great the difference between a device's current clock setting and the correct time.

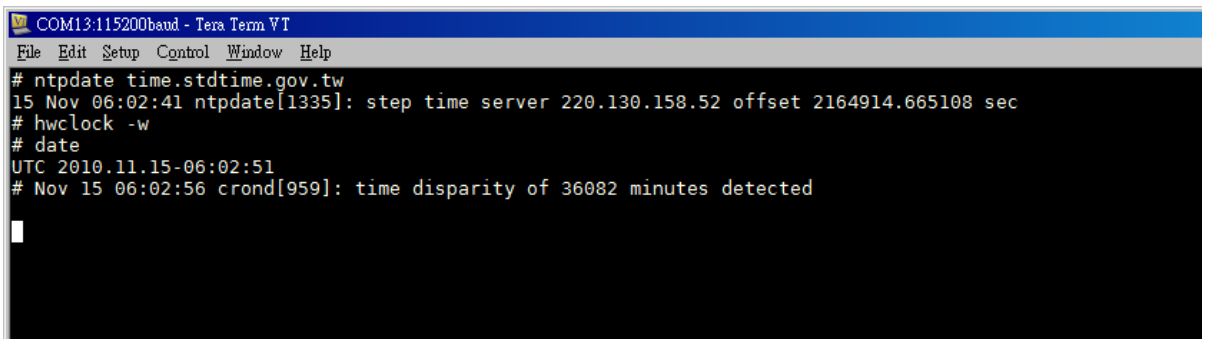
3-5-1 NTP Client

The JetBox has a built-in NTP (Network Time Protocol) client that is used to initialize a time request to a remote NTP server.

Use #**ntpdate** to update the system time.

```
#ntpdate time.stdtime.gov.tw

#hwclock –w
```


COM13:115200baud - Tera Term VT
File Edit Setup Control Window Help
ntpdate time.stdtime.gov.tw
15 Nov 06:02:41 ntpdate[1335]: step time server 220.130.158.52 offset 2164914.665108 sec
hwclock -w
date
UTC 2010.11.15-06:02:51
Nov 15 06:02:56 crond[959]: time disparity of 36082 minutes detected

NTP client request

Visit http://www.ntp.org for more information about NTP and NTP server addresses.

NOTE

Before using the NTP client utility, check your IP and DNS settings to make sure that an Internet connection is available.

3-5-2 NTP Server

NTP is configured by the /etc/ntp.conf file. Here is a simple example:

```
server time.stdtime.gov.tw
driftfile /var/db/ntp.drift
```

The **server** option specifies which servers are to be used, with one server listed on each line. The **driftfile** option specifies which file is used to store the system clock's frequency offset. The ntpd program uses this to automatically compensate for the clock's natural drift, allowing it to maintain a reasonably correct setting even if it is cut off from all external time sources for a period of time.

To start the server, execute the command:

```
# ntpd
```

To check the server status, execute the command:

```
# ntpq -p 127.0.0.1
```

For more information, read the following links about setting up a NTP server:
http://www.freebsd.org/doc/handbook/network-ntp.html

3-6 Cron Daemon (Schedule jobs)

Cron is a daemon to execute scheduled commands. Cron wakes up every minute, examining the /etc/crontab, checking each command to see if it should be run in the current minute.

Crontab syntax :

A crontab file has five fields for specifying day, date and time followed by the command to be run at that interval.

```
*     *     *   *    *        command to be executed
-     -     -   -    -
|     |     |   |    |
```

```
|      |      |    |     +----- day of week (0 - 6) (Sunday=0)
|      |      |    +------- month (1 - 12)
|      |      +---------- day of month (1 - 31)
|      +------------ hour (0 - 23)
+-------------- min (0 - 59)
```

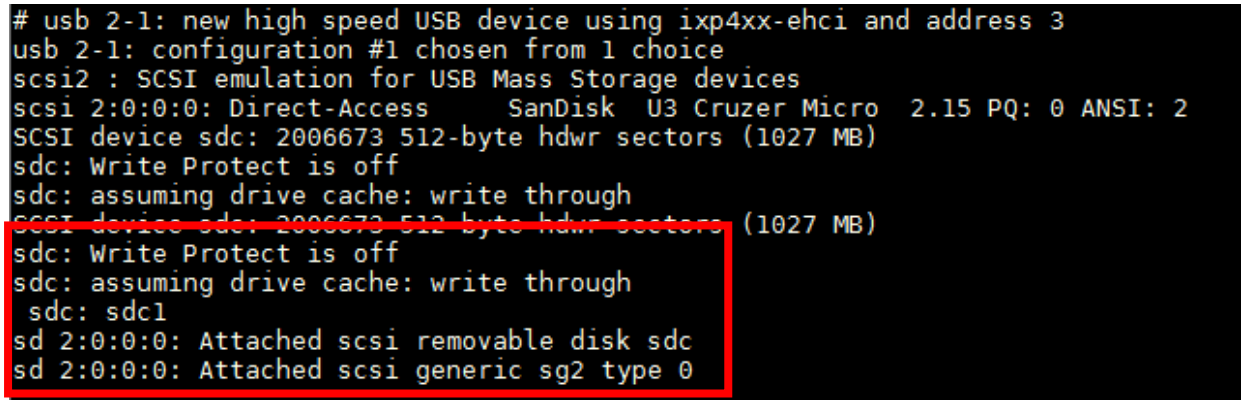
Crontab example :

A line in crontab file like below removes the tmp files from /tmp each day at 6:30 PM.

```
30  18  *  *  *         rm /tmp/*
```

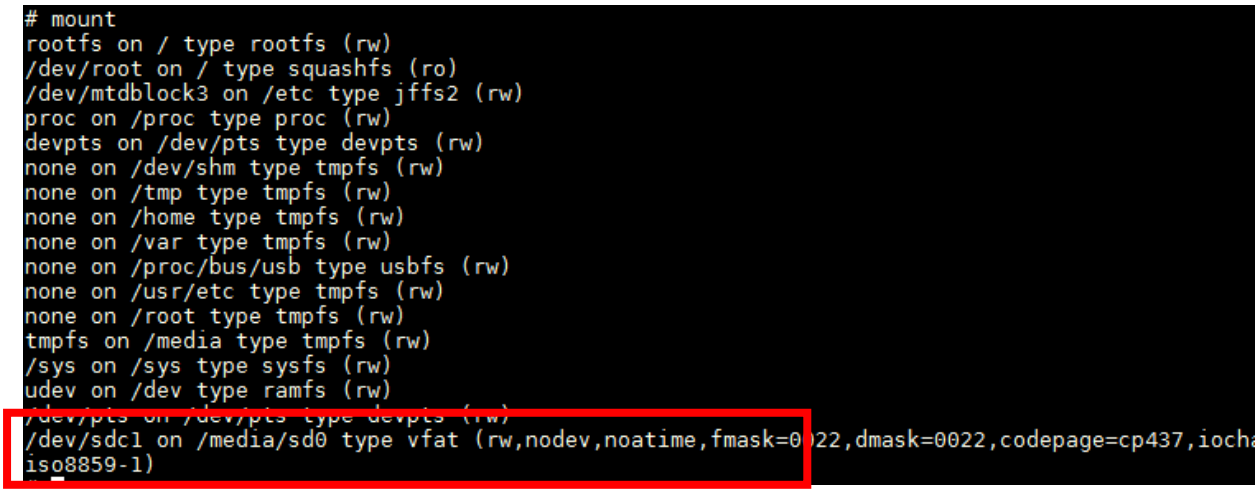
3-7 Connect Peripherals

While plug-in a USB mass storage or a SD card, use **#dmesg** command can help showing USB-storage device status.


usb 2-1: new high speed USB device using ixp4xx-ehci and address 3
usb 2-1: configuration #1 chosen from 1 choice
scsi2 : SCSI emulation for USB Mass Storage devices
scsi 2:0:0:0: Direct-Access   SanDisk U3 Cruzer Micro 2.15 PQ: 0 ANSI: 2
SCSI device sdc: 2006673 512-byte hdwr sectors (1027 MB)
sdc: Write Protect is off
sdc: assuming drive cache: write through
(1027 MB)
sdc: Write Protect is off
sdc: assuming drive cache: write through
sdc: sdc1
sd 2:0:0:0: Attached scsi removable disk sdc
sd 2:0:0:0: Attached scsi generic sg2 type 0

usb-storage device scan status

JetBox9500 support auto-mount external storage, just use mount command to check the mount point.


mount
rootfs on / type rootfs (rw)
/dev/root on / type squashfs (ro)
/dev/mtdblock3 on /etc type jffs2 (rw)
proc on /proc type proc (rw)
devpts on /dev/pts type devpts (rw)
none on /dev/shm type tmpfs (rw)
none on /tmp type tmpfs (rw)
none on /home type tmpfs (rw)
none on /var type tmpfs (rw)
none on /proc/bus/usb type usbfs (rw)
none on /usr/etc type tmpfs (rw)
none on /root type tmpfs (rw)
tmpfs on /media type tmpfs (rw)
/sys on /sys type sysfs (rw)
udev on /dev type ramfs (rw)
/dev/sdc1 on /media/sd0 type vfat (rw,nodev,noatime,fmask=0
22,dmask=0022,codepage=cp437,iocha
iso8859-1)

As the picture shows, the usb-storage has been mounted on /media/sd0. You can access your data in the /media/sd0 folder.

To un-mount the usb-storage, execute #umount <mount path>. For example, issue #umount /media/sd0 can un-mount the previous mounted directory.

NOTE

To be able to unmount a device, you have to close all the open files in it. Type **sync** can help commits all pending writes, which can then be removed in a safe way.

Chapter 4 Network Feature

In this chapter, we explain how to configure JetBox various communication functions.

4-1 Telnet

Service name	telnetd
Description	A Telnet server
Config files	/etc/inetd.conf
Start file	/etc/init.d/rcS
Start command	
Stop command	
Support command	
Default	up

Enabling the Telnet server

The following example shows the default content of the file /etc/inetd.conf. The default is to enable the Telnet server:

```
telnet stream tcp nowait root /usr/sbin/telnetd telnetd -i -l /bin/login
```

Disabling the Telnet server

Disable the daemon by typing '#' in front of the first character of the row to comment out the line.

```
#telnet stream tcp nowait root /usr/sbin/telnetd telnetd -i -l /bin/login
```

4-2 sshd

Service name	sshd
Description	A ssh server
Config files	/etc/sshd_config /etc/ssh_config /etc/ssh_host_dsa_key /etc/ssh_host_dsa_key.pub /etc/ssh_host_key /etc/ssh_host_key.pub /etc/ssh_host_rsa_key /etc/ssh_host_rsa_key.pub
Start file	/etc/init.d/S50sshd

Start command	/etc/init.d/S50sshd start
Stop command	/etc/init.d/S50sshd stop
Support command	
Default	up

Re-generate sshd host keys

The JetBox9500 comes with a set of default sshd host keys. To re-generate it, remove them and restart the ssh daemon.

```
# rm -f /etc/ssh_host_dsa_key /etc/ssh_host_dsa_key.pub /etc/ssh_host_key
/etc/ssh_host_key.pub /etc/ssh_host_rsa_key /etc/ssh_host_rsa_key.pub
# /etc/init.d/S50sshd restart
```

4-3 FTP

Service name	proftpd
Description	A Highly configurable FTP server
Config files	/etc/proftpd.conf
Start file	/etc/init.d/S50proftpd
Start command	/etc/init.d/S50proftpd start
Stop command	/etc/init.d/S50proftpd stop
Support command	
Default	down

Enabling root login

Edit the /etc/proftpd.conf and add the following line, then restart the FTP server.

```
RootLogin on
```

4-4 DNS

To set up DNS client, you need to edit two configuration files:

/etc/resolv.conf,

/etc/hosts (optional)

/etc/hosts - The static table lookup for host names

This is the first file that the Linux system reads to resolve the host name and IP address.

/etc/resolv.conf – DNS resolver configuration file

This is the most important file that you need to edit when using DNS for the other programs. For example, before you use #ntpdate time.nist.goc to update the system time, you will need to add the DNS server address to the file. Ask your network administrator which DNS server address you should use.

The DNS server's IP address is specified with the "nameserver" command. For example, add the following line to /etc/resolv.conf if the DNS server's IP address is 168.95.1.1:

```
nameserver 168.95.1.1
```

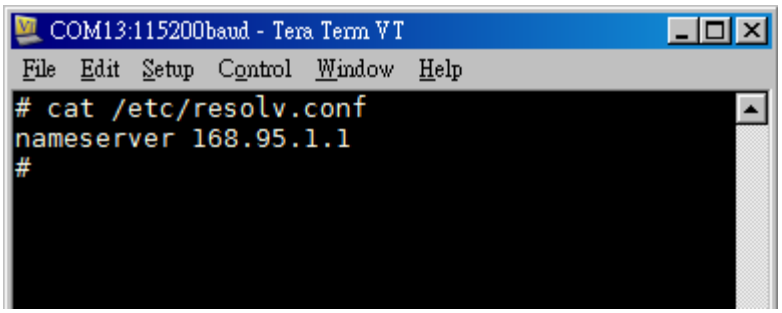


Figure 4-5 nameserver

4-5 Web Server

Service name	webs
Description	web server for embedded system
Config files	
Start file	
Start command	/etc/init.d/S70webs start
Stop command	/etc/init.d/S70webs stop
Support command	webs
Default	down

The GoAhead WebServer is an open standard web server that is compliant with all necessary standards to be an effective embedded web server.

The GoAhead web server's main start file is /etc/init.d/S70webs, with the default homepage located at /web/home.htm. The web server default listening port is 8080. The default Web homepage is in the following directory:

/web

Before you modify the homepage, use a browser (such as Microsoft Internet Explore or Mozilla (Firefox) from your PC to test if the goahead Web Server is working. Type the LAN IP address in the browser's address box to open the homepage.
http://192.168.10.1:8080/home.htm

Because the /web directory is not writable, you can either mount a memory to it:

```
mount –t tmpfs tmpfs /web
```

Or just mount a CF or SD card directory to the /web directory. For example:

```
mount /media/sd0 /web
```

For more information, read the following links about GoAhead web server:
http://www.goahead.com/products/webserver/specifications.aspx

NOTE

Start from firmware v1.3, you can simply change the web directory with the following command:

```
export WEB_HOME=<YOUR WEB DIRECTORY>
```

4-6 IPTABLES

program name	iptables
Description	A NFS client
Usage:	
iptables -[AD] chain rule-specification [options]	
iptables -[RI] chain rulenum rule-specification [options]	
iptables -D chain rulenum [options]	
iptables -[LFZ] [chain] [options]	
iptables -[NX] chain	
iptables -E old-chain-name new-chain-name	
iptables -P chain target [options]	
iptables -h (print this help information)	

iptables is a user space application program on JetBox9500 that allows to configure the tables provided by the Linux kernel firewall (implemented as different Netfilter modules) and the chains and rules it stores. Different kernel modules and programs are currently used for different protocols; iptables applies to IPv4, ip6tables to IPv6, arptables to ARP, and ebtables as a special for Ethernet frames. A firewall using iptables is said to be a stateful firewall.

iptables splits the packet handling into three different tables, each of which contain a number of chains. The firewalling rules, which we create, are included within a particular chain. The three tables are:

1. **filter**: used for packet filtering
2. **nat**: used to provide packet modification capabilities; NAT/PAT and IP masquerading
3. **mangle**: used for setting packet options and marking packets for further filtering or routing

The **filter** table is the default table for any rule. It is where the bulk of the work in an iptables firewall occurs. This table contains three chains:

1. INPUT: used for traffic which is entering our system and belongs to an IP address which is on our local machine
2. OUTPUT: used for traffic which originated on the local system, otherwise known as the firewall
3. FORWARD: used for traffic which is being routed between two network interfaces on our firewall

There are three main targets for a rule within the filter table.

1. ACCEPT: allows the packet to be passed through the firewall without any noticeable interaction
2. DROP: simply drops the packet as if it has never been in the system
3. REJECT: drops the packet then sends a ICMP reply back to the client telling it why the connection failed

Example:

Add rules

The basic syntax of an iptables command is:

```
iptables -A INPUT -s 192.168.20.0/24 -j ACCEPT
```

This would add a rule into the INPUT chain, which matches any packet with a source address in the 192.168.20.0 subnet. If a packet matches this criteria, then it would use the ACCEPT target, which simply allows the packet on through.

Remove rules

To delete the first rule in the chain, we would do:

```
iptables -D INPUT 1
```

List rules

To list the rules we have on our system use:

```
iptables -L
```

Flush rules

To flush (drop) all the rules we can use:

```
iptables -F
```

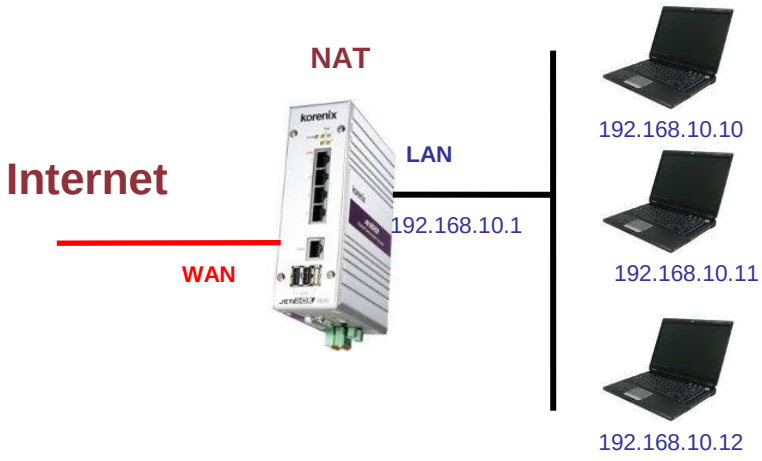
A more complete tutorial can be found at:
http://www.linode.com/wiki/index.php/Netfilter_IPTables_Mini_Howto

4-7 NAT

Network address translation (NAT) is the process of modifying network address information in datagram (IP) packet headers while in transit across a traffic routing device for the purpose of remapping one IP address space into another.

A basic NAT scenario:

The 2 interfaces concerned will be WAN and LAN.
WAN: This will be the interface connected to the Internet.
LAN: This interface will be connected to the private network.



Assuming that you have already configured your system to be able to connect to Internet, run the following command to enable NAT.

```
iptables -t nat -A POSTROUTING -o wan -j MASQUERADE
iptables -A FORWARD -i wan -o lan -m state --state RELATED,ESTABLISHED -j ACCEPT
iptables -A FORWARD -i lan -o wan -j ACCEPT
```

NOTE

If WAN interface use PPPOE to connect to ISP, use **ppp0** interface instead of WAN port.

To save current setting of iptables to a file, use the following command:

```
iptables-save > /etc/iptables-rules
```

To restore the previous saved file, use the following command:

```
iptables-restore /etc/iptables-rules
```

4-8 Dail-up Service

Service name	pppd
Description	Point-to-Point Protocol Daemon
Config files	/etc/options /etc/pap-secrets /etc/chap-secrets
Start file	
Start command	
Stop command	
Support command	**pppd** **chat** **pppdump**
Default	down

PPP (Point to Point Protocol) is used to run IP (Internet Protocol) and other network protocols over a serial link. PPP can be used for direct serial connections (using a null-modem cable) over a Telnet link, and links established using a modem over a telephone line. Usually pppd is called by other daemon, like PPPoE, PPTP, and Wvdial.

4-9 PPPoE

Service name	pppoe
Description	Point-to-Point Protocol over ethernet
Config files	/etc/ppp/pppoe.conf
Start file	

Start command	
Stop command	
Support command	**pppoe-connect** **pppoe-setup** **pppoe-start** **pppoe-stop** **pppoe-status**
Default	down

Point-to-Point Protocol over Ethernet is a network protocol for encapsulating Poing-to-Point Protocol (PPP) frames inside Ethernet frames. It is used mainly with ADSL services where individual users connect to the ADSL transceiver over Ethernet.

You can just inpput pppoe-setup to configure the PPPoE. First you should obtain PPPOE username and password from your Internet provider. Second, filled in these parameters to pppoe-setup dialog:

- **Ethernet Interface: wan**
- **User name: <from your ISP>**
- **Activate-on-demand: No**
- **Primary DNS: 168.95.1.1**
- **Firewalling: NONE**

```
# pppoe-setup
Welcome to the Roaring Penguin PPPoE client setup.  First, I will run
some checks on your system to make sure the PPPoE client is installed
properly...

Looks good!  Now, please enter some information:

USER NAME
>>> Enter your PPPoE user name (default bxxxnxnx@sympatico.ca): 7360011@hinet.net

INTERFACE
>>> Enter the Ethernet interface connected to the DSL modem
(default eth0): wan

Do you want the link to come up on demand, or stay up continuously?
>>> Enter the demand value (default no): no
```

```
DNS

Please enter the IP address of your ISP's primary DNS server.
>>> Enter the secondary DNS server address here:

PASSWORD

>>> Please enter your PPPoE password:
```

Then use the pppoe-start command to start dial to connect network.

4-10 NFS

program name	mount
Description	A NFS client
Usage: mount -t nfs -o nolock NFS_Server_Address:/directory /mount/point	
Example: A NFS server export it's /root directory with IP address 192.168.1.10, want to mount to JetBox's /mnt directory: mount -t nfs -o nolock 192.168.1.10:/root /mnt	

NOTE

Read the following links for more information about setting up a NFS server:

- http://nfs.sourceforge.net/nfs-howto/
- http://nfs.sourceforge.net/nfs-howto/ar01s04.html

4-11 E-Mail

program name	sendmail
Description	A email client
Usage: sendmail [-w timeout] [-U user] [-P password] [-X] -t to [-t to]... [-n] [-s subject] [-c charset] server[:port] from [body] [attachment ...] Options:	

-w timeout	Set timeout on network operations
-U username -P password	Authenticate with specified username/password
-t address	Recipient(s). May be repeated
-X	Use openssl connection helper for secured servers
-n	Request delivery notification to sender
-s subject	Subject
-c charset	Assumed charset for body and subject [utf-8]

Example:

sendmail -U sender@korenix.com -P mypassword -t receiver@korenix.com -s 'A subject' smtp.korenix.com sender@korenix.com /home/mail-content

4-12 SNMP

Service name	snmpd
Description	SNMP Daemon
Config files	/etc/snmp/snmpd.conf
Start file	
Start command	
Stop command	
Support command	snmpget snmpset
Default	down

Net-SNMP is a suite of applications used to implement SNMP v1, SNMP v2c and SNMP v3 using both IPv4 and IPv6. It supports RFC 1213 MIB-II.

For more information, read the following links about NET-SNMP:
http://www.net-snmp.org/wiki/index.php/Tutorials

4-13 OpenVPN

Service name	openvpn
Description	A full-featured SSL VPN
Config files	/etc/openvpn/ /etc/openvpn/easy-rsa/
Start file	/etc/init.d/openvpn

Start command	/etc/init.d/openvpn start
Stop command	/etc/init.d/openvpn stop
Default	down

OpenVPN is a full-featured SSL VPN which implements OSI layer 2 or 3 secure network extension using the industry standard SSL/TLS protocol, supports flexible client authentication methods based on certificates, smart cards, and/or username/password credentials, and allows user or group-specific access control policies using firewall rules applied to the VPN virtual interface.

For more information, download the step-by-step how to from Korenix website:
http://www.korenixembedded.com/support/faqs/vpn

4-14 OpenSwan

Service name	**ipsec**
Description	A full-featured IPSec VPN
Config files	/etc/ipsec.conf
Start file	/etc/init.d/ipsec
Start command	/etc/init.d/ipsec start
Stop command	/etc/init.d/ipsec stop
Default	down

Usage:

ipsec setup [--showonly] {--start|--stop|--restart}

ipsec setup –status

ipsec auto [--showonly] [--asynchronous] --up connectionname

ipsec auto [--showonly] --{add|delete|replace|down} connectionname

ipsec auto [--showonly] --{route|unroute} connectionname

ipsec auto [--showonly] --{ready|status|rereadsecrets|rereadgroups}

ipsec auto [--showonly] --{rereadcacerts|rereadaacerts|rereadocspcerts}

ipsec auto [--showonly] --{rereadacerts|rereadcrls|rereadall}

ipsec auto [--showonly] [--utc] --{listpubkeys|listcerts}

ipsec auto [--showonly] [--utc] --{listcacerts|listaacerts|listocspcerts}

ipsec auto [--showonly] [--utc] --{listacerts|listgroups}

ipsec auto [--showonly] [--utc] --{listcrls|listocsp|listall}

ipsec auto [--showonly] --purgeocsp

OpenSwan is an implementation of IPsec Protocol for Linux. You can create a VPN using ipsec command. ipsec invokes any of several utilities involved in controlling the IPsec encryption/authentication system, running the specified command with the specified arguments as if it had been invoked directly.

For more information, download the step-by-step how to from Korenix website:
http://www.korenixembedded.com/support/faqs/vpn
Or read OpenSwan Wiki:
http://wiki.openswan.org/

4-15 PPTP Client

program name	pptp
Description	A Point-to-Point potocol client
Config file	/etc/ppp/options.pptp /etc/ppp/chap-secrets

Usage:

pptp <hostname> [<pptp options>] [[--] <pppd options>]

Or using pppd's pty option:

pppd pty "pptp <hostname> --nolaunchpppd <pptp options>"

Available pptp options:

--phone <number>	Pass <number> to remote host as phone number
--nolaunchpppd	Do not launch pppd, for use as a pppd pty
--quirks <quirk>	Work around a buggy PPTP implementation Currently recognised values are BEZEQ_ISRAEL only
--debug	Run in foreground (for debugging with gdb)
--sync	Enable Synchronous HDLC (pppd must use it too)
--timeout <secs>	Time to wait for reordered packets (0.01 to 10 secs)
--nobuffer	Disable packet buffering and reordering completely
--idle-wait	Time to wait before sending echo request
--max-echo-wait	Time to wait before giving up on lack of reply
--logstring <name>	Use <name> instead of 'anon' in syslog messages
--localbind <addr>	Bind to specified IP address instead of wildcard
--loglevel <level>	Sets the debugging level (0=low, 1=default, 2=high)

PPTP establishes the client side of a Virtual Private Network (VPN) using the Point-to-Point Tunneling Protocol (PPTP). Use this program to connect to an employer's PPTP based VPN, or to certain cable and ADSL service providers.

Example:

The PPTP Server has the following information:

- The IP address of the server ($SERVER)
- The authentication domain name ($DOMAIN)
- The username you are to use ($USERNAME)
- The password you are to use ($PASSWORD)

In the steps below, substitute these values manually..

Edit the config file /etc/ppp/options.pptp, which sets options common to all tunnels:

```
lock noauth nobsdcomp nodeflate
```

Add authentication information to /etc/ppp/chap-secrets file:

```
# Secrets for authentication
$DOMAIN\\$USERNAME PPTP $PASSWORD *
```

Create a /etc/ppp/peers/$TUNNEL_NAME file.

```
# Secrets for authentication
pty "pptp $SERVER --nolaunchpppd"
name $DOMAIN\\$USERNAME
remotename PPTP
require-mppe-128
file /etc/ppp/options.pptp
ipparam $TUNNEL_NAME
```

Start the tunnel using the pppd command:

```
pppd call $TUNNEL_NAME
```

The following picture is the success output with tunnel name “test” and server ip address “192.168.1.54”.

```
# pppd call test
pppd options in effect:
debug           # (from /etc/ppp/peers/test)
dump            # (from /etc/ppp/options)
noauth          # (from /etc/ppp/options.pptp)
name jim                # (from /etc/ppp/peers/test)
remotename PPTP         # (from /etc/ppp/peers/test)
                # (from /etc/ppp/options.pptp)
pty pptp 192.168.1.54 --nolaunchpppd            # (from /etc/ppp/peers/test)
ipparam test            # (from /etc/ppp/peers/test)
nobsdcomp               # (from /etc/ppp/options.pptp)
nodeflate               # (from /etc/ppp/options.pptp)
require-mppe-128                # (from /etc/ppp/peers/test)
```

To further diagnose a failure, add options to the command:

```
pppd call $TUNNEL_NAME debug dump logfd 2 nodetach
```

NOTE

If you are using a PPTP Server that does not require an authentication domain name, omit the slashes as well as the domain name.

If you need to setup a PPTP server on Windows XP as a test target, please see the following link: http://www.onecomputerguy.com/networking/xp_vpn_server.htm

4-16 PPTP Server

Service name	pptpd
Description	A Point-to-Point potocol server
Config files	/etc/pptpd.conf /etc/ppp/chap-secrets /etc/ppp/options.pptpd
Start file	/etc/init.d/pptpd
Start command	/etc/init.d/pptpd start
Stop command	/etc/init.d/pptpd stop
Default	down

pptpd is the Poptop PPTP daemon, which manages tunneled PPP connections encapsulated in GRE using the PPTP VPN protocol. It may contain features like IP address management and TCP wrappers if compiled in.

Example:

Edit the configuration file: **/etc/pptpd.conf** for IP ranges and option file, for example:

```
option /etc/ppp/options.pptpd
localip 192.168.0.1
remoteip 192.168.0.234-238,192.168.0.245
```

Add user/password lists to **/etc/ppp/chap-secrets**:

```
username pptpd password *
```

Leave the /etc/ppp/options.pptpd unchanged.

Start the tunnel using the command:

```
/etc/init.d/pptpd start
```

4-17 Routing -- Zebra Daemon

Service name	zebra

Description	Zebra is an advanced routing software package that provides TCP/IP based routing protocols.
Config files	/etc/zebra.conf
Start file	/etc/init.d/zebra
Start command	/etc/init.d/zebra start
Stop command	/etc/init.d/zebra stop
Default	down

There are four routing daemons in use, and there is one manager daemon.

- ospfd, ripd, bgpd*, isisd* (* for optional)
- Zebra

Configuration options:

- Each of the daemons has its own config file. For example, zebra's default config file name is /etc/zebra.conf.
- The daemon name plus .conf is the default config file name. You can specify other config file using the -f options when starting the daemon.
- Check the log files for proper operation. For example, you can type less -f /var/log/zebra.log to check the zebra log. Keep in mind that you have to add the log-file location to the respective daemon configuration file.

The administrator has two options to modify runtime configurations via the command-line interface (CLI):

1. (**Preferred**) Use the integrated Zebra shell **vtysh** by typing vtysh. vtysh expects its configuration to reside in /etc/vtysh.conf.
2. Telnet localhost <port> ,e.g. port 2601 connects to the ospfd. Zebra uses ports from 2600 to 2607 for daemon connections.

Service	zebra	ripd	ospfd	bgpd	isisd
Port	2601	2602	2604	2605	2607

Example:

1. Start zebra: /etc/init.d/zebra start
2. Type vtysh to enter Zebra shell

```
# /etc/init.d/zebra start
Starting zebra services: done
# vtysh

Hello, this is Quagga (version 0.99.9).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

JetBox 9530#
```

3. Query for supported command with '?'

```
# vtysh

Hello, this is Quagga (version 0.99.9).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

JetBox 9530#
  clear         Reset functions
  configure     Configuration from vty interface
  copy          Copy from one file to another
  debug         Debugging functions (see also 'undebug')
  disable       Turn off privileged mode command
  end           End current mode and change to enable mode
  exit          Exit current mode and down to previous mode
  list          Print command list
  no            Negate a command or set its defaults
  ping          Send echo messages
  quit          Exit current mode and down to previous mode
  show          Show running system information
  ssh           Open an ssh connection
  start-shell   Start UNIX shell
  telnet        Open a telnet connection
  terminal      Set terminal line parameters
  traceroute    Trace route to destination
  undebug       Disable debugging functions (see also 'debug')
  write         Write running configuration to memory, network, or terminal
JetBox 9530#
```

4. Display the current running config: # sh run

```
JetBox 9530# sh run
Building configuration...

Current configuration:
!
end
hostname Router
!
password zebra
enable password zebra
!
interface dummy0
 ipv6 nd suppress-ra
!
```

5. Enter config mode:

JetBox # **config terminal**

6. Change hostname to KorenixRouter:

JetBox (confg) # hostname KorenixRouter

```
JetBox 9530# configure terminal
JetBox 9530(config)# hostname KorenixRouter
KorenixRouter(config)#
```

For all Zebra suppored command, please refer:
http://www.zebra.org/zebra/Command-Index.html#Command%20Index

4-18 Ser2net

program name	ser2net
Description	ser2net is a Linux program which will connect a network to the serial port. It could be like a bridge between the ethernet cable and the serial cable.
config file	/etc/ser2net.conf

Usage:

-c <config file> - use a config file besides /etc/ser2net.conf

-C <config line> - Handle a single configuration line. This may be specified multiple times for multiple lines. This is just like a line in the config file. This disables the default config file, you must specify a -c after the last -C to have it read a config file, too.

-p <controller port> - Start a controller session on the given TCP port

-P <file> - set location of pid file

-n - Don't detach from the controlling terminal

-d - Don't detach and send debug I/O to standard output

-u - Disable UUCP locking

-b - Do CISCO IOS baud-rate negotiation, instead of RFC2217

-v - print the program's version and exit

Example:

- **Setup a TCP server with following operation parameter:**

Serial port : 1
TCP port : 62001
Baud rate : 9600
Data bits : 8
Parity : none
Stop bit : 1
Hardware flow control : none
State : raw state
timeout : never timeout
modem mode : none

Edit /etc/ser2net.conf add the following line:

```
62001:raw:0:/dev/ttyS1:9600 NONE 1STOPBIT 8DATABITS LOCAL -RTSCTS
```

Then run the ser2net program:

```
ser2net &
```

For more information, please see section 7-7 .

4-19 fastpath

Service name	fastpath
Description	Hardware accelerated packet flow path
Config files	
Start file	
Start command	fastpath start
Stop command	fastpath stop
Default	down

Usage:

start	start fastpath
stop	stop fastpath
-L	list fastpath rules
-A	add fastpath rule [options]
-D rulenum	delete fastpath rule rulenum

Options:

-s	source ip address
-d	destination ip address

Example:

- **Start fastpath:**

fastpath start

FastPath auto ipv4 routing start ...

- **Stop fastpath:**

fastpath stop

FastPath auto ipv4 routing stop ...

- **Add a static route in fastpath with source ip address 192.168.1.1 and destination address 192.168.10.101:**

fastpath -A -s 192.168.1.1 -d 192.168.10.101

- **List current fastpath rules:**

fastpath –L

[root@JetBox 9530 /etc]# fastpath -L

FastPath Show...

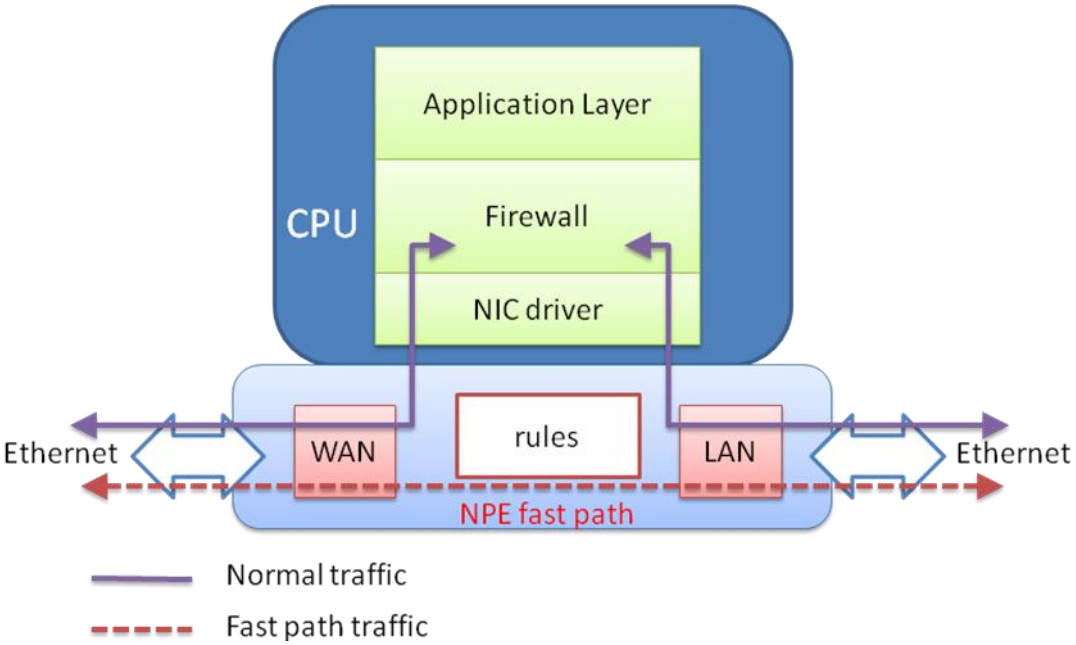
rulenum Lan IP Wan IP

0 192.168.10.101 192.168.1.1

- **Remove a fastpath rule by rule number:**

fastpath -D 0

As the figure bellow, the traffic handled by the JetBox9500 moves through the fast-path alone without any CPU intervention.



This fast-path can receive packets, perform rule look ups, determine the ip address to which the packet should be routed to, and transmit the packet out using hardware engines. Packets that cannot be processed by fast-path routed to the CPU/control-path. As a result, higher level stack services such as HTTP, DHCP, etc. can still be provided via the CPU path while still maintaining throughput performance consistently using the fast-path.

NOTE

- The fast-path database is limit to 16 rules (connections).
- Both source and destination address MUST be reachable; otherwise fastpath can't learn their MAC address.

4-20 keepalived

Service name	keepalived
Description	keepalived implements an independent VRRPv2 stack to handle director failover. It can deal with the Linux Virtual Server (LVS) framework.
Config files	/etc/keepalived/keepalived.conf

Start file	
Start command	
Stop command	
Default	down

Usage:

```
  keepalived
  keepalived -n
  keepalived -f keepalived.conf
  keepalived -d
  keepalived -h
  keepalived -v

Commands:
Either long or short options are allowed.
  keepalived --vrrp              -P    Only run with VRRP subsystem.
  keepalived --check             -C    Only run with Health-checker subsystem.
  keepalived --dont-release-vrrp  -V    Dont remove VRRP VIPs & VROUTEs on daemon stop.
  keepalived --dont-release-ipvs -I    Dont remove IPVS topology on daemon stop.
  keepalived --dont-fork         -n    Dont fork the daemon process.
  keepalived --use-file          -f    Use the specified configuration file.
                            Default is /etc/keepalived/keepalived.conf.
  keepalived --dump-conf         -d    Dump the configuration data.
  keepalived --log-console       -l    Log message to local console.
  keepalived --log-detail        -D    Detailed log messages.
  keepalived --log-facility      -S    0-7 Set syslog facility to LOG_LOCAL[0-7].
(default=LOG_DAEMON)
  keepalived --help              -h    Display this short inlined help screen.
  keepalived --version           -v    Display the version number
  keepalived --pid               -p    pidfile
  keepalived --checkers_pid       -c    checkers pidfile
  keepalived --vrrp_pid          -r    vrrp pidfile
```

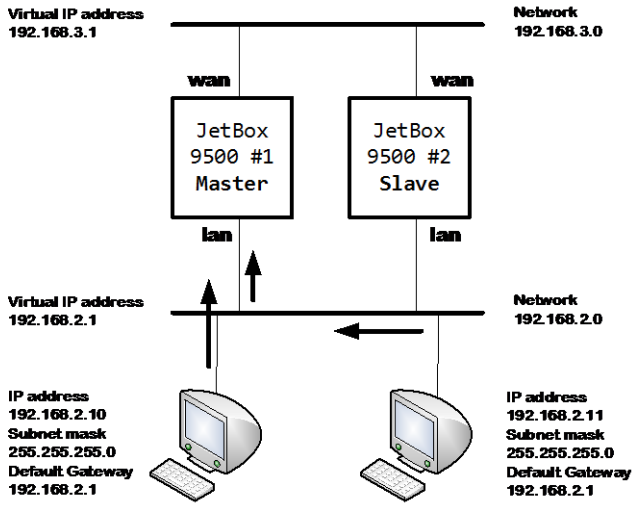
Example:

- **Start keepalived with VRRP subsystem only:**

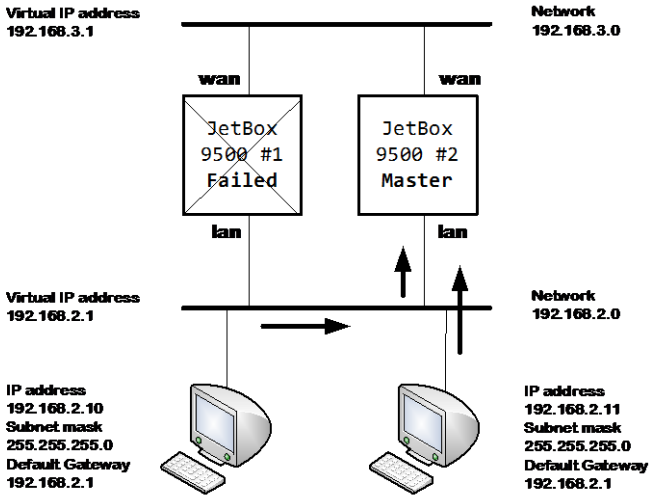
keepalived -P

Virtual Router Redundancy Protocol (VRRP) is an IETF standard that provides a redundant router mechanism to increase the reliability of existing networks. The VRRP process provides

dynamic failover of the forwarding responsibility to a Backup (secondary) router if the Master (primary) router becomes unavailable. VRRP provides the advantage of a single default path that has higher availability without requiring network devices to be reconfigured if a failure occurs.



The VRRP protocol provides rapid transition from Master to Backup to minimize service interruption.
One of the JetBox becomes the Master and the other becomes a Backup when the VRRP protocol starts up. Election of the Master is based on the configuration file. Both Master and Backup have identical Virtual IP Addresses.



When Master becomes unavailable, traffic between JetBox 9500 #1 and JetBox 9500 #2 is interrupted. Then JetBox 9500#2 will take the place as Master.

JetBox9500 #1	JetBox9500 #2

vrrp_sync_group VG2 {
group {
VI_WAN
VI_LAN
}
}
	vrrp_sync_group VG2 {
	group {
	VI_WAN
	VI_LAN
	}
	}

vrrp_instance VI_WAN { state MASTER interface wan virtual_router_id 151 priority 150 advert_int 1	vrrp_instance VI_WAN { state BACKUP interface wan virtual_router_id 151 priority 100 advert_int 1

authentication {
auth_type PASS
auth_pass 1111
}
virtual_ipaddress {
192.168.3.1/24 brd
192.168.3.255 dev wan
}
}
	authentication {
	auth_type PASS
	auth_pass 1111
	}
	virtual_ipaddress {
	192.168.3.1/24 brd
	192.168.3.255 dev wan
	}
	}

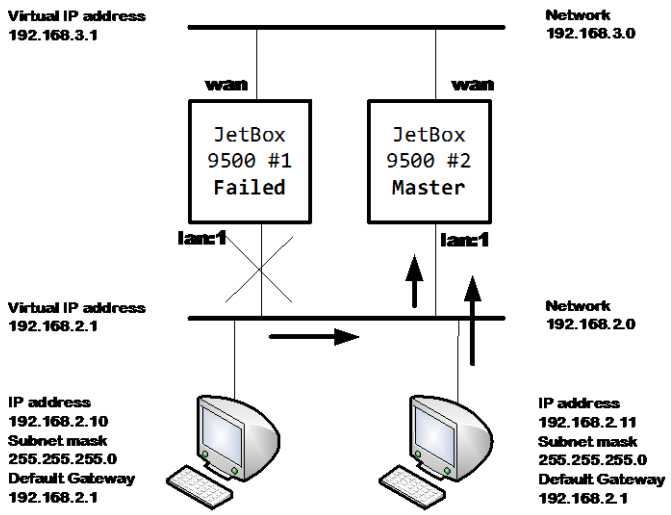
vrrp_instance VI_LAN {
state MASTER
interface lan
virtual_router_id 152
priority 150
advert_int 1
authentication {
auth_type PASS
auth_pass 1111
}
virtual_ipaddress {
192.168.2.1/24 brd
192.168.2.255 dev lan
}
}
	vrrp_instance VI_LAN {
	state BACKUP
	interface lan
	virtual_router_id 152
	priority 100
	advert_int 1
	authentication {
	auth_type PASS
	auth_pass 1111
	}
	virtual_ipaddress {
	192.168.2.1/24 brd
	192.168.2.255 dev lan
	}
	}

Keyword	Definition	Type
vrrp_sync_group	make sure that several router instances stay together on a failure	
vrrp_instance	identify a VRRP instance definition block	
state	specify the instance state in standard use	
interface	specify the network interface for the instance to run on	string
virtual_router_id	specify to which VRRP router id the instance belongs	numerical
priority	specify the instance priority in the VRRP router	numerical
advert_int	specify the advertisement interval in seconds (set to 1)	numerical
smtp_alert	Activate the SMTP notification for MASTER state transition	
authentication	identify a VRRP authentication definition block	
auth_type	specify which kind of authentication to use (PASS|AH)	
auth_pass	specify the password string to use	string
virtual_ipaddress	identify a VRRP VIP definition block	
notify_master	specify a shell script to be executed during transition to master state	path
notify_backup	specify a shell script to be executed during transition to backup state	path
notify_fault	specify a shell script to be executed during transition to fault state	path

The **linkbeat_use_polling** keyword

When device become unavailable, both wan and lan port will stop send vrrp packet. But if you

need to monitor link status, you will need to use linkbeat_use_polling keyword.



The following configuration tells keepalived to monitor lan interface port 1's link status. When JetBox #1 lan port 1 link is down, both wan and lan will turn into FAULT state. Thus JetBox #2 will turn into MASTER state, and route packet between two interfaces.

JetBox9500 #1	JetBox9500 #2

linkbeat_use_polling
vrrp_sync_group VG2 {
group {
VI_WAN
VI_LAN
}
}
	linkbeat_use_polling
	vrrp_sync_group VG2 {
	group {
	VI_WAN
	VI_LAN
	}
	}

vrrp_instance VI_WAN {
state MASTER
interface wan
virtual_router_id 151
priority 150
advert_int 1
authentication {
auth_type PASS
auth_pass 1111
}
virtual_ipaddress {
192.168.3.1/24 brd
192.168.3.255 dev wan
}
}
	vrrp_instance VI_WAN {
	state BACKUP
	interface wan
	virtual_router_id 151
	priority 100
	advert_int 1
	authentication {
	auth_type PASS
	auth_pass 1111
	}
	virtual_ipaddress {
	192.168.3.1/24 brd
	192.168.3.255 dev wan
	}
	}

vrrp_instance VI_LAN {
state MASTER
interface lan:1
virtual_router_id 152
priority 150
advert_int 1
authentication {
auth_type PASS
auth_pass 1111
}
virtual_ipaddress {
192.168.2.1/24 brd
	vrrp_instance VI_LAN {
	state BACKUP
	interface lan:1
	virtual_router_id 152
	priority 100
	advert_int 1
	authentication {
	auth_type PASS
	auth_pass 1111
	}
	virtual_ipaddress {
	192.168.2.1/24 brd

192.168.2.255 dev lan } }	192.168.2.255 dev lan } }

NOTE

- The default port for linkbeat_use_polling is port 1
- keepalived currently not support Virtual MAC address

4-21 PTPD

Service name	ptpd
Description	Precision Time protocol (PTP) version 1 daemon
Config files	
Start file	
Start command	ptpd
Default	down

Usage: ptpd [OPTION]

```
-c                run in command line (non-daemon) mode
-f FILE           send stats to FILE
-S                     send output to syslog
-T                set multicast time to live
-d                display stats
-D                display stats in .csv format
-R                record data about sync packets in a file

-x                do not reset the clock if off by more than one second
-O                do not reset the clock if offset is more than NUMBER nanoseconds
-M                do not accept delay values of more than NUMBER nanoseconds
-t                do not adjust the system clock
-a NUMBER,NUMBER  specify clock servo P and I attenuations
-w NUMBER         specify one way delay filter stiffness

-b NAME           bind PTP to network interface NAME
-u ADDRESS        also send uni-cast to ADDRESS
-l NUMBER,NUMBER  specify inbound, outbound latency in nsec

-o NUMBER         specify current UTC offset
```

```
-e NUMBER         specify epoch NUMBER
-h                specify half epoch

-y NUMBER         specify sync interval in 2^NUMBER sec
-m NUMBER         specify max number of foreign master records

-g                run as slave only
-p                make this a preferred clock
-s NUMBER         specify system clock stratum
-i NAME           specify system clock identifier
-v NUMBER         specify system clock allen variance

-n NAME           specify PTP subdomain name (not related to IP or DNS)

-k NUMBER,NUMBER  send a management message of key, record, then exit
```

Example:

- **Start ptpd in slave mode:**

ptpd -g

- **Stop ptpd in master mode (preferred clock):**

ptpd -p

The PTP daemon (PTPd) implements the Precision Time protocol (PTP) as defined by the relevant IEEE 1588 standard. PTP Version 1 implements IEEE-1588-2002, and PTP Version 2 implements IEEE-1588-2008. PTP was developed to provide very precise time coordination of LAN connected computers.

4-22 PTPDv2

Service name	ptpd2
Description	Precision Time protocol (PTP) version 2 daemon
Config files	
Start file	
Start command	ptpd2
Default	down

Usage:

```
Ptpv2d runs on UDP/IP , P2P mode by default

-c                run in command line (non-daemon) mode
-f FILE           send output to FILE
```

```
-S                send output to syslog
-T               set multicast time to live
-d               display stats
-D               display stats in .csv format
-R               record data about sync packets in a file

-x               do not reset the clock if off by more than one second
-O               do not reset the clock if offset is more than NUMBER nanoseconds
-M               do not accept delay values of more than NUMBER nanoseconds
-t               do not adjust the system clock
-a NUMBER,NUMBER  specify clock servo P and I attenuations
-w NUMBER         specify one way delay filter stiffness

-b NAME           bind PTP to network interface NAME
-u ADDRESS        also send uni-cast to ADDRESS
-e               run in ethernet mode (level2)
-h               run in End to End mode
-l NUMBER,NUMBER  specify inbound, outbound latency in nsec

-o NUMBER         specify current UTC offset
-i NUMBER         specify PTP domain number

-n NUMBER         specify announce interval in 2^NUMBER sec
-y NUMBER         specify sync interval in 2^NUMBER sec
-m NUMBER         specify max number of foreign master records

-g               run as slave only
-v NUMBER         specify system clock allen variance
-r NUMBER         specify system clock accuracy
-s NUMBER         specify system clock class
-p NUMBER         specify priority1 attribute
-q NUMBER         specify priority2 attribute
```

Example:

- **Start ptpd2 in slave mode:**

ptpd2 -g

- **Start ptpd2 in master mode :**

ptpd2

The PTP daemon (PTPd) implements the Precision Time protocol (PTP) as defined by the

relevant IEEE 1588 standard. PTP Version 1 implements IEEE-1588-2002, and PTP Version 2 implements IEEE-1588-2008. PTP was developed to provide very precise time coordination of LAN connected computers.

Chapter 5 Korenix Feature

5-1 Ethertool

program name	ethtool
Description	LAN switch port configuration tool

Usage:

ethtool DEVNAME (get port status)

ethtool -a DEVNAME (get flow control)

ethtool -A DEVNAME (set flow control)

[rx on|off]

ethtool -q DEVNAME (get QoS)

ethtool -Q DEVNAME (set QoS)

[type 0|1] (0: weighted, 1: Strict)

[trust 0-4] (0:Port Based, 1:CoS Only, 2:DSCP Only, 3:CoS First, 4:DSCP First)

[pri 0-7] (Port default frame priority)

[cos QUEUE_ID COS_ID]

[dscp QUEUE_ID DSCP_ID]

ethtool -p DEVNAME (get vlan pvid)

ethtool -P DEVNAME (set vlan pvid)

[pvid N]

ethtool -t DEVNAME (reset statistic)

ethtool -s DEVNAME (set port status)

[speed 10|100]

[duplex half|full]

[port on|off] - Enable or Disable this port

[autoneg on|off]

ethtool -S DEVNAME (get statistic)

Example:

- **Get LAN interface port 3 status**

ethtool lan:3
Port Status:
Speed: 100Mb/s
Duplex: Full
Flow Control: off
Port Setting:
Port: on
Auto-negotiation: on
Link detected: Down

- **Set LAN interface port 3 status**

ethtool -s lan:3 speed 100

ethtool -s lan:3 port off
ethtool -s lan:3 autoneg off
ethtool -s lan:3 duplex half

- **Set LAN interface port 3 vlan pvid to 2**

ethtool -P lan:3 pvid 2

5-2 Vconfig

program name	vconfig
Description	Create and remove VLAN devices on LAN port

Usage:

Options:

add	[interface-name] [vlan_id] tag [port_id] untag [port_id]
	[example : vconfig add lan 2 tag 1,2,3 untag 4]
rem	[vlan-name]
show	[Display Vlan Table]
set_flag	[interface-name] [flag-num] [0 | 1]
set_egress_map	[vlan-name] [skb_priority] [vlan_qos]
set_ingress_map	[vlan-name] [skb_priority] [vlan_qos]
set_name_type	[name-type]

Example:

- **Add a vlan interface**

vconfig add lan 2 untag 1,2,3,4

```
lan       Link encap:Ethernet  HWaddr 00:12:77:50:04:CB
          inet addr:192.168.10.1  Bcast:192.168.10.255  Mask:255.255.255.0
          inet6 addr: fe80::212:77ff:fe50:4cb/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:33 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1073 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:512
          RX bytes:3318 (3.2 KiB)  TX bytes:86421 (84.3 KiB)

lan.2     Link encap:Ethernet  HWaddr 00:12:77:50:04:CB
          BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

- **Delete a vlan interface**

vconfig rem lan.2

- **Show all vlan interface**

vconfig show

```
//----------------------------------------//
VLAN ID        : 1
Tag Port       :
Un-Tag Port : 1, 2, 3, 4,
```

NOTE

When you add a VLAN interface, you can only see it by typing “ifconfig -a”. It will display with lan.2”.

To enable it, issue “ifconfig lan.2 up

5-3 Diocfg

program name	diocfg
Description	Get/Set DIO Configuration

Usage:

diocfg -g PORT_NUM - get DIO status

diocfg -s PORT_NUM [0|1] - set DO [OFF|ON]

diocfg -T PORT_NUM [0|1] - set DIO Type [Output|Input]

diocfg -S Save Config

diocfg -r Read Config

diocfg -c Clean Config

Example:

- **Get status and type on port 7**

diocfg -g 7
Get data of DI7 to High

- **Set DIO type to DO on port 7**

diocfg -T 7 0
Set Port 7 to Ouput

- **Set DO 7 to High**

diocfg -s 7 1
Set data of DO7 to High

NOTE

The DIO number range is 0~7. The DIO types are configurable. Default all DIO type are set with DI.

5-4 Poecfg

program name	poecfg
Description	Get/Set POE Device Configuration

Usage:

poecfg -g PORT_NUM - get PoE status

poecfg -s PORT_NUM [0|1] - set PoE [OFF|ON]

poecfg -S Save Config

poecfg -r Read Config

poecfg -c Clean Config

Example:

- **Get POE status on port 1 (without PD on port 1)**

```
# poecfg -g 1
PoE Port 1
        control status is On
        power delivery status is Off
```

- **Get POE status on port 4 (PD on port 4, and power is 48 voltage)**

```
# poecfg -g 4
PoE Port 4
        control status is On
        power delivery status is On
        Voltage 46.2V
        Current 74mA
        Power 3.41W
```

- **Turn off POE on port 1**

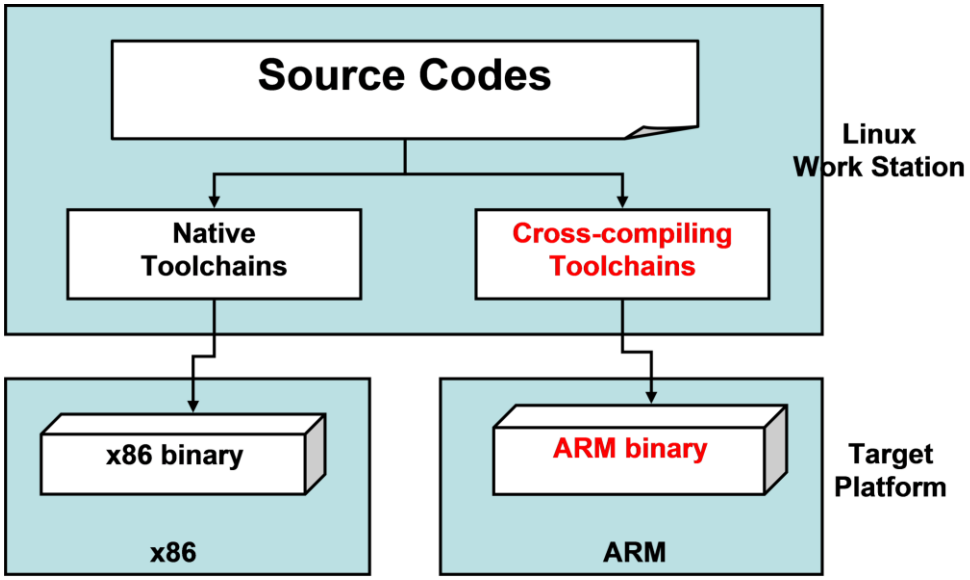
```
# poecfg -s 1 0
Set PoE Port 1 to Off
# poecfg -g 1
PoE Port 1
        control status is Off
        power delivery status is Off
```

Chapter 6 Programmer's Guide

6-1 Toolchain Introduction

To ensure that an application will be able to run correctly when installed on Jetbox, you must ensure that it is compiled and linked to the same libraries that will be present on the Jetbox9500.

The cross-compiling toolchain that comes with Jetbox9500 contains a suite of Korenix compilers and other tools, as well as the libraries and headers that are necessary to compile applications for Jetbox9500. The build environment must be running Linux and install with the Jetbox9500 Toolchain. We have confirmed that the following Linux distributions can be used to install the tool chain: Fedora Core 9, Centos 5.



The Toolchain will need about 208 MB of hard disk space on your Linux PC. The Jetbox9500 toolchain is included in the JetBox9500 SDK, which can download from http://www.korenixembedded.com.

The SDK can be extract at any directory, for example, your HOME directory.
You can extract the SDK with following command:

```
tar zxvf jetbox9500-sdk-<version>.tar.gz
```

```
[Fedora 9] $ls
ap_src/  environment.sh*  jetbox9500-toolchain.tar.bz2  README
```

The README file will teach you how to install the Toolchain, and application examples are in the **ap_src** directory.

6-1-1 Compiling Applications and Libraries

To compile a simple C application, just use the aremb-linux-gcc compiler instead of the regular one:

```
armeb-linux-gcc –g -02 source-code.c –o output
```

6-1-2 Tools Available in the Host Environment

The following cross compiler tools are provided:

armeb-linux-ar	Manage archives (static libraries)
armeb-linux-as	Assembler
armeb-linux-c++	C++ compiler
armeb-linux-cpp	C preprocessor
armeb-linux-g++	C++ compiler
armeb-linux-gcc	C compiler
armeb-linux-gprof	Display call graph profile data
armeb-linux-ld	Linker
armeb-linux-nm	Lists symbols from object files
armeb-linux-objcopy	Copies and translates object files
armeb-linux-objdump	Displays information about object files
armeb-linux-ranlib	Generates indexes to archives (static libraries)
armeb-linux-readelf	Displays information about ELF files
armeb-linux-size	Lists object file section sizes
armeb-linux-strings	Prints strings of printable characters from files (usually object files)
armeb-linux-strip	Removes symbols and sections from object files (usually debugging information)

6-2 Device API

ioctl

Name

ioctl - control device

Library

Standard C Library (libc, -lc)

Synopsis

#include <sys/ioctl.h>

int **ioctl**(*int d*, *unsigned long request*, ...);

Description

The **ioctl**() system call manipulates the underlying device parameters of special files. In particular, many operating characteristics of character special files (e.g. terminals) may be controlled with **ioctl**() requests. The argument *d* must be an open file descriptor.

The third argument to **ioctl**() is traditionally named *char *argp*. Most uses of **ioctl**(), however, require the third argument to be a *caddr_t* or an *int*.

An **ioctl**() *request* has encoded in it whether the argument is an ''in'' argument or ''out'' argument, and the size of the argument *argp* in bytes. Macros and defines used in specifying an ioctl *request* are located in the file <*sys/ioctl.h*>.

Return Values

If an error has occurred, a value of -1 is returned and *errno* is set to indicate the error.

Please use the desktop Linux's man page for detailed documentation:
#man ioctl

6-3 RTC

The device node is located at /dev/rtc0. Jetbox supports Linux standard simple RTC control.
You must include <linux/rtc.h>

.

1. Function: RTC_RD_TIME
   int ioctl(fd, RTC_RD_TIME, struct rtc_time *time);
   Description: read time information from RTC. It will return the value on argument 3.

2. Function: RTC_SET_TIME
   int ioctl(fd, RTC_SET_TIME, struct rtc_time *time);
   Description: set RTC time. Argument 3 will be passed to RTC.

6-4 Watch Dog Timer

A Watchdog Timer (WDT) is a hardware circuit that can reset the computer system in case of a software fault. You probably knew that already.

The Watchdog Driver has one basic role: to talk to the card and send signals to it so it doesn't reset your computer ... at least during normal operation.

The ioctl API:

- Pinging the watchdog using an ioctl:
  WDIOC_KEEPALIVE:,
  This ioctl does exactly the same thing as a write to the watchdog device, so the main loop in the program could be:

```
while (1) {
        ioctl(fd, WDIOC_KEEPALIVE, 0);
        sleep(10);
}
```

  The argument to the ioctl is ignored.

- Setting and getting the timeout:

  To modify the watchdog timeout on the fly with the SETTIMEOUT ioctl, driver has the WDIOF_SETTIMEOUT flag set in their option field. The argument is an integer representing the timeout in seconds. The driver returns the real timeout used in the same variable, and this timeout might differ from the requested one due to limitation of the hardware.

```
int timeout = 45;
ioctl(fd, WDIOC_SETTIMEOUT, &timeout);
printf("The timeout was set to %d seconds\n", timeout);
```

Starting with the Linux 2.4.18 kernel, it is possible to query the current timeout using the GETTIMEOUT ioctl.

```
ioctl(fd, WDIOC_GETTIMEOUT, &timeout);
printf("The timeout was is %d seconds\n", timeout);
```

Chapter 7 Appendix

7-1 Firmware Upgrade

Firmware upgrade can be done by the “firmware-up” command.

```
# firmware-up


Usage: -f [firmware file]
       -t [firmware file] [tftp server]
       -w [http or ftp url]
```

For example, put the new firmware in a USB storage and plug-in to JetBox9500. After it automatic mounted, you can upgrade with the following command:

```
firmware-up -f /media/sd0/Jetbox95-20100916.bin
```

Where /media/sd0 is the usb mounted directory, and Jetbox95-20100916.bin is the firmware name.

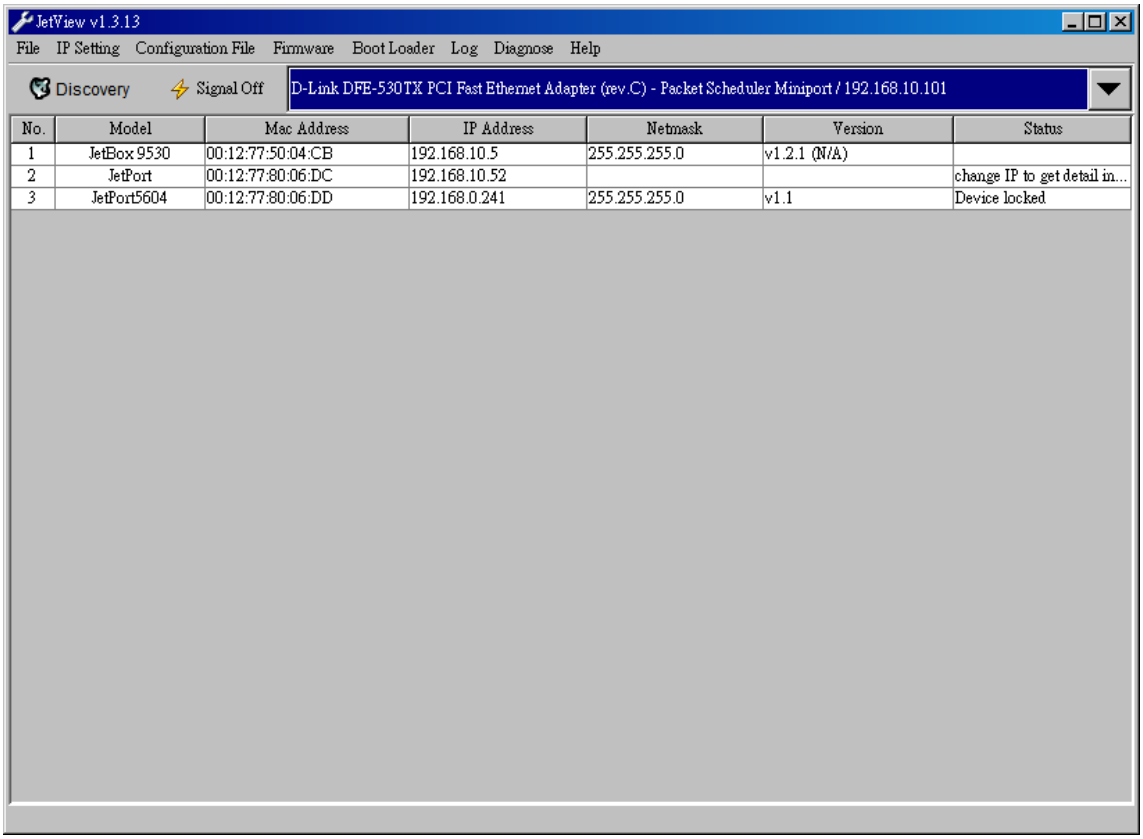
7-2 JetView

7-2-1 Overview

The JetView is a device management utility which support various device management features- such as device recovery, firmware and boot loader upgrade, configuration backup and restore, system event log listing, basic system IP address modify.

7-2-2 JetView for JetBox 9500

Start from firmware v1.3, JetBox9500 can be discovered on LAN port, change IP address, firmware upgrade, open web GUI, and reboot through JetView.


JetView v1.3.13
File IP Setting Configuration File Firmware Boot Loader Log Diagnose Help
Discovery
Signal Off
D-Link DFE-530TX PCI Fast Ethernet Adapter (rev.C) - Packet Scheduler Miniport / 192.168.10.101
No. Model Mac Address IP Address Netmask Version Status
1 JetBox 9530 00:12:77:50:04:CB 192.168.10.5 255.255.255.0 v1.2.1 (N/A)
2 JetPort 00:12:77:80:06:DC 192.168.10.52 change IP to get detail in...
3 JetPort5604 00:12:77:80:06:DD 192.168.0.241 255.255.255.0 v1.1 Device locked

7-3 USB Driver for GSM modems

Some USB 3G modems use the driver USB_SERIAL_COPTION in linux kernel. However, not all user specified USB idProduct and idVendor are include in the driver. For USB idProduct and idVendor not included in linux kernel v2.6.20, JetOS has other way to add it.

```
modprobe c_option vendor=idVender product=idProduct
```

7-4 Change RS485 mode on JetCard 1608

Before you start using JetCard 1608 RS-422/485, configure the DIP Switch settings of each port. DIP Switch can configure RS-422, 4-wire RS-485, or 2-wire RS-485. In addition to configure the DIP Switch to setup hardware, use **setserial** command to tell the JetCard driver to change RS-485 mode is necessary. (Only RS-485 need to invoke setserial command)

RS485 2W

```
setserial -a /dev/ttyS1 mode 485_2W_ON card 1608
setserial -a /dev/ttyS1 mode 485_2W_OFF card 1608
```

RS485 4W

```
setserial -a /dev/ttyS1 mode 485_4W_ON card 1608
setserial -a /dev/ttyS1 mode 485_4W_OFF card 1608
```

JetCard 1608 Select the Serial Interface (RS-232/422/485) for Com1

RS-232 (Default)					RS-422			
Port1 (SW1)	OFF	ON	OFF	OFF	ON	ON	ON	ON
Port2 (SW2)	OFF	ON	OFF	OFF	ON	ON	ON	ON
Port3 (SW3)	OFF	ON	OFF	OFF	ON	ON	ON	ON
Port4 (SW4)	OFF	ON	OFF	OFF	ON	ON	ON	ON

RS-485/4W					RS-485/2W			
Port1 (SW1)	ON	OFF	ON	ON	OFF	OFF	ON	OFF
Port2 (SW2)	ON	OFF	ON	ON	OFF	OFF	ON	OFF
Port3 (SW3)	ON	OFF	ON	ON	OFF	OFF	ON	OFF
Port4 (SW4)	ON	OFF	ON	ON	OFF	OFF	ON	OFF

7-5 Software Specification

Item	Protocol	Notes	JetBox 9500
Boot Loader			Red-boot
Kernel			2.6.20
	ARP		x
	PPP		x
	CHAP		x
	IPv4		x
	IPv6		x
	PAP		x
	ICMP		x
	TCP		x
	UDP		x
	NFS		x
File System			
JFFS2			x
NFS			x
Ext2			x

Item	Protocol	Notes	JetBox 9500
Ext3			x
VFAT			x
FAT			x
Base SW package			
Shell		OS shell command	GNU ash
Busybox		Linux normal command utility	1.11.1
telnetd	Telnet	telnet server daemon	x
inetd		TCP server manager program	x
udhcp	DHCP	DHCP client/server	x
syslogd			x
e2fsprogs		Ext2/Ext3 file system utilities	1.39
i2c-tools	I2C	I2C tools for Linux	3.0.1
ltp-testsuite		Linux Test Project	20070228
microcom		Serial port terminal	1.02
mtd		MTD/JFFS2 utilities	1.1.0
pciutils	PCI	PCI utilities	2.2.10
setserial		RS-232 serial port setting tool	2.17
usbmount	USB	automatically mounts USB mass storage devices	0.0.14.1
usbutils	USB	USB utilities	0.72
Network related SW package			
bridge-utils		Ethernet bridge utility	1.0.6
ethtool		Ethernet configure tool	6
iptables		NAT setting tool	1.3.8
net-snmp	SNMP v1/v2c/v3	SNMP support package	5.1.2
ntp	NTP	NTP utility	4.2.4p2
openssh	SSH1.0/2.0	SSH support package	4.6p1
openssl	SSL	SSL support package	0.9.7m
openvpn	OpenVPN	VPN tool	2.0.9
openswan	IPsec	Ipsec for Linux	2.4.9
pppd	PPP	PPP protocol for Linux	2.4.4
rp-pppoe	PPPoE	PPPOE support package	3.8
pptp-linux	PPTP	PPTP protocol for Linux	1.7.0

Item	Protocol	Notes	JetBox 9500
proftpd	FTP	FTP daemon	1.3.1
samba		SMB (Windows network) support package	3.0.28a
goahead	HTTP	Web server	2.5
sendmail		email client	
bind	DNS	DNS server	9.3.2
l2tp	L2TP	L2TP protocol for Linux	0.70
mrouted	DVMRP	DVMRP multicast routing protocol	3.9b3
quagga	OSPFv1.0/2.0, RIPv1.0/2.0/ng, BGP4, ISIS	unicast routing protocol	0.99.9
wireless-tools	802.11	Tools of WLAN card	29
Graphic and Multimedia			
jamvm		JAVA Virtual Machine	1.5.0
Linux tool chain			
Gcc		C/C++ PC Cross Compiler	4.2.4
uClibc		POSIX standard C library	0.9.29
Performance			
BogoMIPS			665.19

7-6 Busybox command

busybox(V1.8.2): Linux command collection

File Manager	
cp	copy file
ls	list file
ln	make symbolic link file
mount	mount and check file system
rm	delete file
chmod	change file owner & group & user
chown	change file owner
chgrp	change file group
sync	Sync file system, let system file buffer be saved to hardware
mv	move file

pwd	display now file directly
df	list now file system space
mkdir	make new directory
rmdir	delete directory

Editor	
vi	text editor
cat	dump file context
zcat	compress or expand files
grep	search string on file
cut	get string on file
find	find file where are there
more	dump file by one page
test	test file exist or not
sleep	sleep(seconds)
echo	Echo string
awk	Pattern scanning and processing language.
diff	compare two files or directories
sed	perform text transformations on a file or input from a pipeline.
xargs	execute a specified command on every item from standard input.

Archival Utilities	
bzip2/bunzip2	Compress/Uncompress bzip FILE
cpio	Extract or list files from a cpio archive
gzip/gunzip	Compress/Uncompress FILE with maximum compression.
tar	Create, extract, or list files from a tar file
unzip	Extract files from ZIP archives

System logging	
syslogd	Utility used to record logs of all the significant events
klogd	Utility which intercepts and logs all messages from the Linux kernel and sends to the 'syslogd'
logger	Utility to send arbitrary text messages to the system log

Network	
ping	ping to test network
arp	Manipulate the system ARP cache
arping	Ping host by ARP packets
ftpget	Retrieve a remote file via FTP
ftpput	Store a remote file via FTP
nslookup	Tool to query Internet name servers
pscan	Simple network port scanner
traceroute	Utility to trace the route of IP packets
wget	Utility for non-interactive download of files from HTTP, HTTPS, and FTP servers.
udhcpc	DHCP client
route	routing table manager
netstat	display network status
Ifconfig	set ip address and configure network interfaces
traceroute	trace route
tftp	Trivial File Transfer Protocol client
telnet	Telnet client
ftp	FTP client

Others	
dmesg	dump kernel log message
stty	stty is used to change and print terminal line settings
zcat	dump .gz file context
mknod	make device node
free	display system memory usage
date	print or set the system date and time
env	run a program in a modified environment
clear	clear the terminal screen
reboot	reboot / power off/on the server
halt	halt the server
du	estimate file space usage
hostname	show system's host name
kill/killall	Send specified signal to the specified process or process group

For complete command usage and explanation, please refer to following website:

http://www.busybox.net/downloads/BusyBox.html

7-7 Ser2net Manual

Name

ser2net - Serial to network proxy

Synopsis

ser2net [-c configfile] [-C configline] [-p controlport] [-n] [-d] [-b] [-v] [-P pidfile]

Description

The **ser2net** daemon allows telnet and tcp sessions to be established with a unit's serial ports.

The program comes up normally as a daemon, opens the TCP ports specified in the configuration file, and waits for connections. Once a connection occurs, the program attempts to set up the connection and open the serial port. If another user is already using the connection or serial port, the connection is refused with an error message.

Options

-c config-file

Set the configuration file to one other than the default of **/etc/ser2net.conf**

-C config-line

Handle a single configuration line. This may be specified multiple times for multiple lines. This is just like a line in the config file. This disables the default config file, you must specify a -c after the last -C to have it read a config file, too.

-n

Stops the daemon from forking and detaching from the controlling terminal. This is useful for running from init.

-d

Like -n, but also sends the system logs to standard output. This is most useful for debugging purposes.

-P pidfile

If specified, put the process id (pid) of ser2net in the pidfile, replacing whatever was in that file previously. A pidfile is not created by default, you must specify this to create one. Note also that this filename must be specific with the full path, as ser2net will change directory to "/" when it becomes a daemon. when it

-u

If UUCP locking is enabled, this will disable the use of UUCP locks.

-b

Cisco IOS uses a different mechanism for specifying the baud rates than the mechanism described in RFC2217. This option sets the IOS version of setting the baud rates. The default is RFC2217's.

-v

Prints the version of the program and exits.

-p controlport

Enables the control port and sets the TCP port to listen to for the control port. A port number may be of the form [host,]port, such as 127.0.0.1,2000 or localhost,2000. If this is specified, it will only bind to the IP address specified for the port. Otherwise, it will bind to all the addresses on the machine.

If the port number is zero, that means that standard in/out will be used for the only input/output, and only one port should be specified in the config. This way, it can be used from inetd.

Control Port

The control port provides a simple interface for controlling the ports and viewing their status. To accomplish this, it has the following commands:

showport [<TCP port>]

Show information about a port. If no port is given, all ports are displayed.

showshortport [<TCP port>]

Show information about a port, each port on one line. If no port is given, all ports are displayed. This can produce very wide output.

help

Display a short list and summary of commands.

exit

Disconnect from the control port.

version

Display the version of this program.

monitor <type> <tcp port>

Display all the input for a given port on the calling control port. Only one direction may be monitored at a time. The type field may be *tcp* or *term* and specifies whether to monitor data from the TCP port or from the serial port Note that data monitoring is best effort, if the controller port cannot keep up the data will be silently dropped. A controller may only monitor one thing and a port may only be monitored by one controller.

monitor stop

Stop the current monitor.

disconnect <tcp port>

Disconnect the tcp connection on the port.

setporttimeout <tcp port> <timeout>

Set the amount of time in seconds before the port connection will be shut down if no activity has been seen on the port.

setportconfig <tcp port> <config>

Set the port configuration as in the device configuration in the **/etc/ser2net.conf** file. If conflicting options are specified, the last option will be the one used. Note that these will not change until the port is disconnected and connected again.

Options *300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200* set the various baud rates. *EVEN, ODD, NONE* set the parity. *1STOPBIT, 2STOPBITS* set the number of stop bits. *7DATABITS,8DATABITS* set the number of data bits. *[-]XONXOFF* turns on (off) XON/XOFF support. *[-]RTSCTS* turns on (- off) hardware flow control. *[-]LOCAL* ignores (- checks) the modem control lines (DCD, DTR, etc.)

setportcontrol <tcp port> <controls>

Modify dynamic port controls. These do not stay between connections. Controls are: *DTRHI, DTRLO* Turns on and off the DTR line. *RTSHI, RTSLO* Turns on and off the RTS line.

setportenable <tcp port> <enable state> Sets the port operation state. Valid states are: *off* to shut the TCP port down, *raw* to enable the TCP port transfer all I/O as-is, *rawlp* to enable the TCP port input and device output without termios setting, and *telnet* to enable the TCP port is up run the telnet negotiation protocol on the port.

Configuration

Configuration is accomplished through the file **/etc/ser2net.conf**. A file with another name or path may be specified using the *-c* option, or individual config lines may be specified with the *-C* option. This file consists of one or more entries with the following format:

<TCP port>:<state>:<timeout>:<device>:<options>

or

BANNER:<banner name>:<banner text>

FIELDS

TCP port

Name or number of the TCP/IP port to accept connections from for this device. A port number may be of the form [host,]port, such as 127.0.0.1,2000 or localhost,2000. If this is specified, it will only bind to the IP address specified for the port. Otherwise, it will bind to all the ports on the machine.

state Either **raw** or **rawlp** or **telnet** or **off.** *off* disables the port from accepting

connections. It can be turned on later from the control port. *raw* enables the port and transfers all data as-is between the port and the long. *rawlp* enables the port and transfers all input data to device, device is open without any termios setting. It allow to use /dev/lpX devices and printers connected to them. *telnet* enables the port and runs the telnet protocol on the port to set up telnet parameters. This is most useful for using telnet.

timeout

The time (in seconds) before the port will be disconnected if there is no activity on it. A zero value disables this funciton.

device The name of the device to connect to. This must be in the form of **/dev/<device>**.

device configuration options

Sets operational parameters for the serial port. Values may be separated by spaces or commas. Options *300,1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200* set the various baud rates. *EVEN, ODD, NONE* set the parity. *1STOPBIT, 2STOPBITS* set the number of stop bits. *7DATABITS, 8DATABITS* set the number of data bits.*[-]XONXOFF* turns on (- off) XON/XOFF support. *[-]RTSCTS* turns on (- off) hardware flow control. *[-]LOCAL*ignores (- checks) the modem control lines (DCD, DTR, etc.) *[-]HANGUP_WHEN_DONE* lowers (- does not lower) the modem control lines (DCD, DTR, etc.) when the connection closes. *NOBREAK* Disables automatic clearing of the break setting of the port. *rem_ctl* allows remote control of the serial port parameters via RFC 2217. See the README for more info. *<banner name>* displays the given banner when a user connects to the port.

banner name

A name for the banner; this may be used in the options of a port.

banner text

The text to display as the banner. This may contain normal "C" escape strings, and it may also contain, \d for the device name, \p for the TCP port number, and \s for the serial port parameters (eg 9600N81) of the given connection.

Blank lines and lines starting with '#' are ignored.

Security

ser2net uses the tcp wrappers interface to implement host-based security.

See hosts_access(5) for a description of the file setup. Two daemons are used by ser2net, "ser2net" is for the data ports and "ser2net-control" is for the control ports.

Signals

SIGHUP

If ser2net receives a SIGHUP, it will reread it configuration file and make the appropriate changes. If an inuse port is changed or deleted, the actual change will not occur until the port

is disconnected.

Error

Almost all error output goes to syslog, not standard output.

Files

/etc/ser2net.conf

See Also

telnet(1) , hosts_access(5)

Known Problems

None.

Author

Corey Minyard <minyard@acm.org>

7-8 Revision history

V0.0.9 by 2011/3/8 update for JetOS93 v1.3 release

7-9 Customer Service

Korenix Technology Co., Ltd.
Business service: sales@korenixembedded.com, sales@korenix.com
Customer service: koreCARE@korenix.com
Web Site: http://www.korenixembedded.com, http://www.korenix.com